Títol: *Mineria de textos extrets de Facebook i Twitter a través dels seus APIs*

Alumne: *Ivan Bargalló Vaca*

Titulació: *Enginyeria informàtica*

Director/Ponent: *Ferran Sabaté Garriga*

Codirector: *Antonio Cañabate Carmona*

Departament: *Organització d'empreses*

Índex

1 Introducció

1.1 Descripció del projecte

Amb el naixement d'Internet i l'explosió de les xarxes socials, en ple 2015, un gran nombre d'empreses que vol millorar la seva presència a Internet, utilitzen xarxes com Facebook i Twitter com un canal més en el seu pla de comunicació i màrqueting.

El temps en el que les empreses es limitaven a oferir un producte als clients mitjançant els canals tradicionals, sense tenir en compte la resposta dels seus consumidors ha passat. Ara moltes empreses estan orientat gradualment els seus serveis i productes a les necessitats i opinions dels clients finals, ja no només per ajustar-se millor a les seves necessitats, sinó també per millorar la imatge de la marca oferint atenció personalitzada a aquests clients.

Els dos punts anteriors han produït el naixement de la figura del Community Manager, és a dir, la persona o grups de persones encarregades de les comunicacions directes amb els clients a través de les pàgines de les marques a les xarxes socials.

Aquestes comunicacions no haurien de ser arbitràries o improvisades, sinó que s'han d'alienar en la política de comunicació de l'empresa i ser especialment dissenyades per aconseguir una resposta positiva dels consumidors.

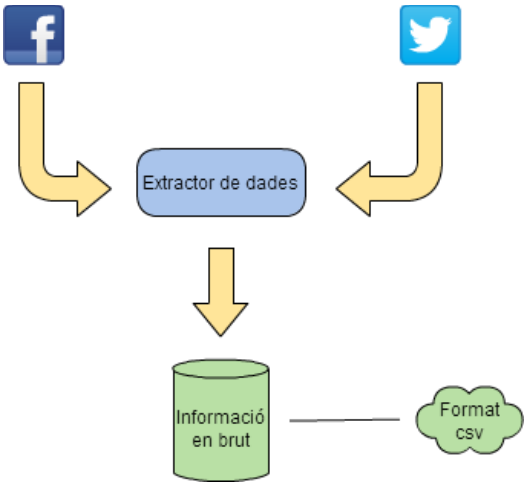
Per tal de millorar les comunicacions esmentades, es desenvolupen contínuament estudis que analitzen la relació que hi ha entre les diverses variables observables a partir de les publicacions que les marques realitzen a les xarxes socials i les reaccions dels seus seguidors o fans. Entre aquestes variables n'hi ha moltes fàcilment quantificables, com comptadors, hores, dates, tipus de contingut, etc... i altres més difícils de quantificar per l'anàlisi posterior i que estan directament relacionades amb el significat i semàntica dels continguts publicats, ja siguin textos, imatges, vídeos i/o enllaços, etc...

Aquest projecte neix amb la intenció d'oferir una sèrie d'eines per a facilitar la realització d'aquests estudis i així poder respondre amb més facilitat als interrogants que els hi sorgeixen a les empreses a la hora de realitzar les seves comunicacions a la xarxa.

D'aquesta manera, el projecte es pot dividir en dos àrees:

- **Extracció de dades a les xarxes socials de Twitter i Facebook**: S'ha desenvolupat una aplicació per tal de poder realitzar la extracció de dades de les xarxes esmentades. El tipus de dades a extreure varia segons l'estudi a realitzar, per aquest motiu l'aplicació creada té cert nivell de personalització per tal de satisfer el major ventall d'opcions possible. La imatge 1 mostra un esquema del funcionament d'aquesta àrea, l'extractor de dades captura la informació de les xarxes socials i les ofereix en format CSV.
- **Anàlisi de textos extrets de les diverses pàgines o comptes de les marques en diferents xarxes socials com Twitter i Facebook**: S'ha investigat la possibilitat d'incorporar Machine Learning per poder classificar els textos extrets amb l'eina esmentada al punt anterior. L'objectiu d'aquesta investigació és estudiar el rendiment dels algoritmes de classificació existents en aquest àmbit al utilitzar com a dades d'entrada textos extrets directament de les xarxes socials. Les conclusions d'aquesta

investigació permetrien conèixer la possibilitat de sistematitzar l'anàlisi dels textos extrets, enriquint d'aquesta manera l'estudi de les relacions entre característiques de les publicacions, els resultats i les reaccions dels usuaris.. La imatge 2 mostra el funcionament general d'aquesta àrea, les dades capturades de l'extractor són netejades i separades en dos blocs de dades per tal d'utilitzar-los en els algoritmes de classificació de Machine Learning, el funcionament concret serà detallat en apartats posteriors.

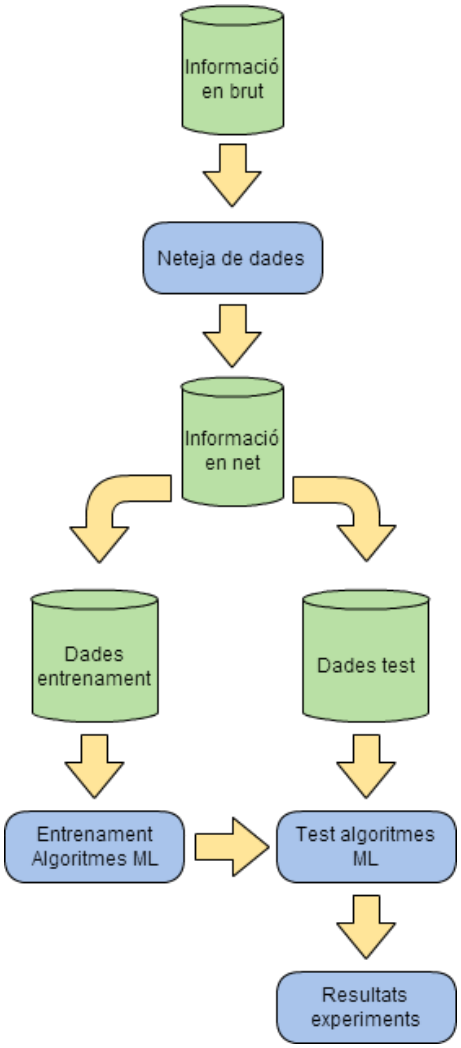


Imatge 1. Esquema de l'àrea d'extracció de dades

1.2 Objectius del projecte

Els objectius que s'han definit per en aquest projecte han sigut els següents:

- Obtenir un mètode per extraure les dades de les xarxes socials amb relativa facilitat per tal de realitzar estudis de recerca.
- Estudiar les bases del Machine Learning classificatori.
- Estudiar la precisió dels algoritmes de Machine Learning principals entrenats amb un conjunt de dades ja definit.
- Concloure la viabilitat de la utilització d'aquests algoritmes en estudis de recerca sobre les xarxes socials.



Imatge 2. Esquema de funcionament de l'àrea de Machine Learning

2 Abast i metodologia del projecte

2.1 Abast

Al començar aquest projecte sense cap tipus de coneixement sobre la temàtica del Machine Learning i l'estat de les aplicacions d'extracció de dades actualment, s'ha hagut d'anar corregint l'abast del projecte segons les diverses problemàtiques anaven sorgint durant aquest.

En la primera part del projecte, la extracció de dades en Twitter i Facebook, consistia en fer una cerca sobre possible programari que complissin les necessitats requerides per possibles estudis d'investigació. Malauradament, la cerca ha sigut infructuosa, trobant greus carències en software gratuït, oferint aquests un mínim conjunt de dades completament insuficient. D'aquesta forma, s'ha hagut de desenvolupar una aplicació des de cero per tal d'obtenir la màxima varietat de dades possibles que no ofereixen aquestes aplicacions de software gratuïtes.

Per altra banda, l'àmbit del Machine Learning és extremadament ampli i complex, oferint una sèrie de possibilitats massa nombroses com per abastar-les en un sol projecte. Per aquest motiu el projecte s'ha centrat en el Machine Learning amb aprenentatge supervisat, concretament, en la classificació de textos mitjançant algoritmes entrenats amb un conjunt de dades classificades prèviament.

Tot i això, en cap moment s'ha descartat de forma absoluta la recerca en altres camps a més del de la classificació de textos També s'ha fet una mínima recerca superficial sobre conceptes com el *clustering* en el camp del Machine Learning amb aprenentatge no supervisat, o fins i tot en el camp de la descripció d'imatges, els quals tant Google com Stanford han obtingut bons resultats últimament. Aquests conceptes i els resultats de la recerca són mostrats a l'apartat de 'Estat de l'art' del present document.

Per concloure, en referència a l'anàlisis i comparació dels diferents algoritmes de classificació, tot i que n'existeixi un nombre molt elevat d'aquests, s'ha limitat el nombre d'algoritmes a estudiar als quatre més esmentats i amb millors resultats de diferents articles revisats utilitzant la implementació oferta per l'aplicació de Machine Learning WEKA.

2.2 Metodologia

Tot i que en els apartats corresponents s'explicarà el procés seguit de forma detallada, en els punts següents es descriurà de forma més genèrica el procés general que s'ha seguit per realitzar el projecte, separat per àrees per tal de fer una lectura més organitzada.

2.2.1 Lectura d'estudis previs

La primera fase del projecte va consistir en la lectura d'un seguit d'articles de recerca de màrqueting a les xarxes socials per tal de tenir un punt d'entrada a aquesta temàtica i entendre quina necessitat es requeria cobrir des del punt de vista dels articles que investiguen la relació entre les característiques de les publicacions de les marques a les xarxes socials i les reaccions dels seus seguidors. Articles com *Online engagement factors on Facebook brand* (Pletikosa i Michahelles et al., 2013) o *Factors influencing popularity of branded content in*

Facebook fan Pages (Sabaté et al., 2014) han sigut una de les principals fonts d'informació disponibles per tal de superar aquesta primera barrera d'entrada.

2.2.2 Organització del projecte

Totes les tasques han sigut coordinades en les diverses reunions de seguiment amb els dos tutors, aportant aquests la direcció que havia de portar el projecte, focalitzant-lo en certes temàtiques, proporcionant articles d'investigació, suggeriments, consells diversos i suport segons la situació ho demanés.

Addicionalment, s'han anat redactant alguns documents amb les troballes realitzades per tal d'aprofitar aquesta documentació com a suport de la memòria final i així no haver de redactar aquesta completament des d'un inici.

2.2.3 Estudi d'aplicacions d'extracció de dades

Un cop obtingut un coneixement inicial sobre la temàtica del màrqueting digital i podent entrar ja pròpiament en el projecte, el primer pas va ser realitzar una reunió amb els tutors per tal de conèixer els requisits necessaris que s'haurien de complir per tal d'obtenir unes dades satisfactòries de les xarxes socials. Amb aquests requisits clars, es va procedir a realitzar una cerca de les aplicacions actuals existents amb la capacitat de complir la tasca demanada, tot i que malauradament, sembla no existir en el mercat actual, una aplicació gratuïta per tal d'obtenir dades mitjanament complexes més enllà d'un nom d'usuari i els posts que s'han escrit a la pàgina de les marques. Un resum de les aplicacions trobades es podrà trobar en l'apartat 3.1 de la memòria.

Coneixent la realitat de la situació actual, es va optar per la creació d'una aplicació pròpia amb la qual es permetrien extreure les dades directament de les xarxes socials.

Per aquest motiu, es va procedir al estudi i documentació de les APIs (Interfície de Programació d'Aplicacions), per tal de coneixes les possibilitats i limitacions que oferien aquestes i així poder planificar la futura aplicació.

El primer pas per realitzar la nova aplicació va consistir en provar alguna llibreria de programació relacionada amb les APIs de Twitter i Facebook per veure les seves capacitats de personalitzar les dades obtingudes amb la màxima facilitat possible.

2.2.4 Creació de l'aplicació d'extracció de dades

Per l'eina d'extracció de dades s'ha tingut en compte els dos requisits principals, el primer que l'eina havia d'extreure dades de les APIs de forma bastant personalitzable i que la programació havia de ser el suficientment senzilla per tal de facilitar modificacions posteriors. El segon requisit tot i ser el més abstracte s'ha intentat realitzar aplicant el paradigma de orientació a objectes amb una arquitectura de tres capes per tal de separar clarament les funcionalitats i així facilitar una lectura posterior del codi. Aquests requisits s'han complert utilitzant llibreries amb suficient capacitat de personalització per tal d'obtenir exactament les dades desitjades i facilitar la modificació del codi en un futur cas de produir-se algun canvi en les APIs de les xarxes.

2.2.5 Estudi sobre Machine Learning

En l'àrea de Machine Learning, es va realitzar una cerca per saber-ne quin tipus era el més adequat per classificar textos, utilitzant com a fonts diversos estudis proporcionats tant pels tutors, com per aquells trobats mitjançant el cercador Google Scholar, donant més importància a aquells documents que han tingut més citacions.

A partir d'aquest moment es va treballar en cercar i aprendre el funcionament dels algoritmes comunament utilitzats en aquests estudis . Un cop estudiats alguns dels exemples recomanats, es va descobrir l'eina WEKA, una aplicació que proveeix un gran conjunt d'algoritmes de Machine Learning amb els quals es podia realitzar la experimentació sobre aquests sense haver de desenvolupar-los un a un de forma manual. Aquesta troballa ha estalviat una cerca més profunda de cadascun dels algoritmes i ha ampliat la quantitat d'algoritmes possibles a provar. L'interès principal del projecte era saber quins resultats podíem obtenir de la seva aplicació, no conèixer el desenvolupament d'aquests en profunditat.

2.2.6 Creació de datasets

Com que els algoritmes de Machine Learning amb aprenentatge supervisat, requereixen utilitzar un conjunt de dades o *dataset,* ja classificades per entrenar-los, es va procedir a crear dos conjunts en format ARFF, el format que llegeixen els algoritmes de WEKA, per tal de realitzar les proves de precisió demanades en el projecte.

El primer dataset va ser proveït per Irena Pletikosa, autora d'un dels articles ja esmentats en aquest apartat, gràcies a la intervenció d'Antonio Cañabate. Aquestes dades consistia en un seguit de post extrets de les pàgines de les xarxes socials, classificats manualment en post de tipus afecte, compartir informació, peticions d'aquesta o suggeriments.

El segon dataset creat ha sigut d'elaboració personal, obtenint els posts mitjançant les primeres versions creades de l'extractor de dades de les APIs i classificant aquests en post de tipus informatiu sobre el producte o empresa, post que oferien algun tipus d'incentiu econòmic a l'usuari, sigui en format de concurs o en format d'ofertes i finalment post que no tenien una relació directa amb l'empresa, utilitzats habitualment com entreteniment per l'usuari, bé per algun tipus d'acudit, comentari sobre algun esdeveniment recent o fins i tot per desitjar un bon cap de setmana als seguidors de la pròpia pàgina.

Cada dataset ha sigut dividit en dos parts, una d'entrenament per servir d'entrada als algoritmes, sent aquesta de 450 pel conjunt de Pletikosa, o bé de 1000 entrades pel conjunt de creació pròpia i un altra part de proves, amb 139 entrades i 200 respectivament.

Un cop obtinguda la separació, es va procedir a la neteja, és a dir, esborrar símbols estranys, enllaços a pàgines web, etc... i a separar cada dataset d'entrenament en múltiples de 100 entrades per tal de comprovar si existia algun punt mitjançant el qual l'adició d'entrades deixava de ser eficient en relació amb la precisió obtinguda dels experiments.

2.2.7 Experimentació

La experimentació posterior ha consistit en provar els diversos datasets creats amb els algoritmes de més utilització en els estudis, canviant paràmetres en cadascuna de les proves

per tal de millorar l'eficiència d'aquests. No només s'han utilitzat els conjunts de proves per tal de saber la eficiència dels algoritmes, també s'han utilitzat alguns dels mètodes que proporciona WEKA, com el *cross-validation*, per tal d'obtenir altres mesures. A partir d'aquí s'han realitzat una sèrie de gràfics per tal de mostrar els resultats dels experiments i documentar aquests en la memòria del projecte.

2.2.8 Redacció de la memòria

Finalment, utilitzant tots els coneixements obtinguts i mitjançant el suport dels documents escrits anteriorment, s'ha redactat la memòria del projecte de forma iterativa, entregant parts d'aquesta per tal d'anar obtenint correccions i comentaris dels tutors, fins al moment d'arribar a l'entrega final d'aquesta.

3 Estat de l'art

3.1 Extracció de dades de les API de xarxes socials

La primera gran àrea del projecte consistia en conèixer les possibilitats que existien actualment per tal d'extraure dades de les xarxes socials.

En els següents apartats es mostraran algunes de les aplicacions d'extracció trobades i les seves característiques bàsiques, les conclusions arribades un cop analitzades aquestes i per finalitzar, un resum de les capacitats que tenen tant Twitter com Facebook per oferir les dades de les seves xarxes a través de les seves APIs.

La intenció de proporcionar aquesta informació es proveir als lectors d'una visió general de les possibilitats existents a l'actualitat en aquesta àrea i de les raons per les quals s'ha decidit crear una aplicació des de cero.

3.1.1 Aplicacions d'extracció de dades existents

A continuació es mostraran alguna de les aplicacions trobades per tal d'extraure dades de les xarxes i les seves característiques bàsiques.

3.1.1.1 TAGS

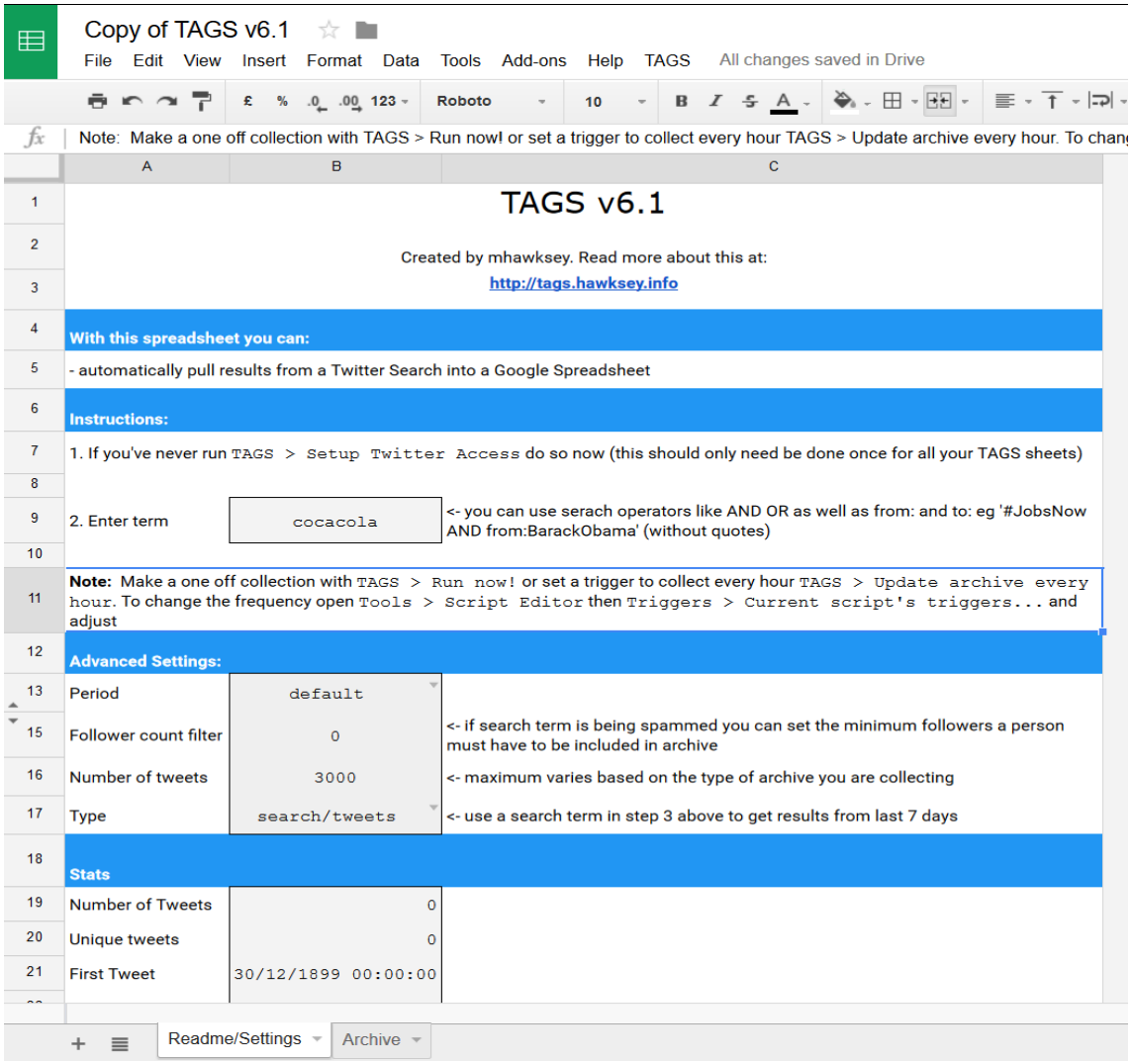
TAGS neix com una opció per descarregar tweets de la xarxa social Twitter mitjançant un add-on per Google Spreadsheets, la versió de Google de l'eina de càlcul Microsoft Excel.

Amb una senzilla interfície, tal com es pot veure a la imatge 1, aquesta aplicació permet l'extracció de tweets utilitzant tres opcions diferents:

- Posts de la pàgina d'un usuari: S'extreuen els tweets de la pàgina principal d'un usuari.
- Cerca de Twitter: S'utilitzen les capacitats que ofereix la Search API de Twitter, la qual se'n parlarà en un apartat posterior, per tal de cercar tweets que compleixen un determinat filtratge.
- Tweets preferits: Es capturen els tweets marcats com a preferits per l'usuari indicat en el camp de cerca.

Per cadascuna d'aquestes opcions, TAGS ofereixen algunes opcions addicionals per filtrar aquestes cerques, com poden ser el nombre de tweets a retornar, el període, en nombre de dies d'antelació a cercar i finalment el nombre mínim de seguidors que ha de tenir un usuari per tal d'aparèixer al llistat de resultats.

Les imatges 2 i 3, mostren el llistat de resultats que ofereix aquest full de càlcul. La informació mostrada és realment completa, mostrant cada apartat en una columna diferent.



Imatge 3. Interfície principal de TAGS

id_str	from_user	text	created_at	time	geo_coordinates	user_lang	in_reply_to_user	in_reply_to_screer
775776	alihr77	RT @auronplay: ola soy un emoji me llamo Antonio jeje i voi a matar a toda tu familia mientras duermes jajaja me gusta la cocacola https://...	Tue Sep 13 19:20:2	13/09/2016 20:20:22		es		
775776	THUNDERPOOL	looking for my stapler. (at @CocaCola Headquarters in Atlanta, GA) https://t.co/UYDhtYMZED	Tue Sep 13 19:20:2	13/09/2016 20:20:21	loc: 33.77020167,-84.39735926	en		
775776	PobleteGustavo	RT @camilanesas: Re insisten con la publicidad de cocacola del q hace magia con las botellas son el fourcade de las publicidades	Tue Sep 13 19:20:0	13/09/2016 20:20:06		es		
775776	CarmenMLJ_03	RT @auronplay: ola soy un emoji me llamo Antonio jeje i voi a matar a toda tu familia mientras duermes jajaja me gusta la cocacola https://...	Tue Sep 13 19:19:5	13/09/2016 20:19:52		es		
775776	Valecillaa	@BolliStuff el pisco es mucho mejor... Con 3 hielos y cocacola... #paraiso	Tue Sep 13 19:19:4	13/09/2016 20:19:47		es	75315591	BolliStuff

Imatge 4. Llistat de resultats de TAGS (1)

from_user_id_str	in_reply_to_status_i	source	profile_image_url	user_followers_coun	user_friends_count	user_location	status_url	entities_str
2280077331		<a href="http://twitte	http://pbs.twimg.com	401	194	♉	http://twitter.com/alih	{"hashtags":[],"symbo
19407886		<a href="http://foursc	http://pbs.twimg.com	1431	2596	atlanta	http://twitter.com/TH	{"hashtags":[],"symbo
864079147		<a href="http://twitte	http://pbs.twimg.com	251	160	Fueguino, Ushuaia	http://twitter.com/Pol	{"hashtags":[],"symbo
711966613273231360		<a href="http://twitte	http://pbs.twimg.com	9	29	Región de Murcia, E	http://twitter.com/Ca	{"hashtags":[],"symbo
85950245	775773063698718	<a href="http://twitte	http://pbs.twimg.com	558	653	Concepción, Chile	http://twitter.com/Val	{"hashtags":[{"text":"p

Imatge 5. Llistat de resultats de TAGS (2)

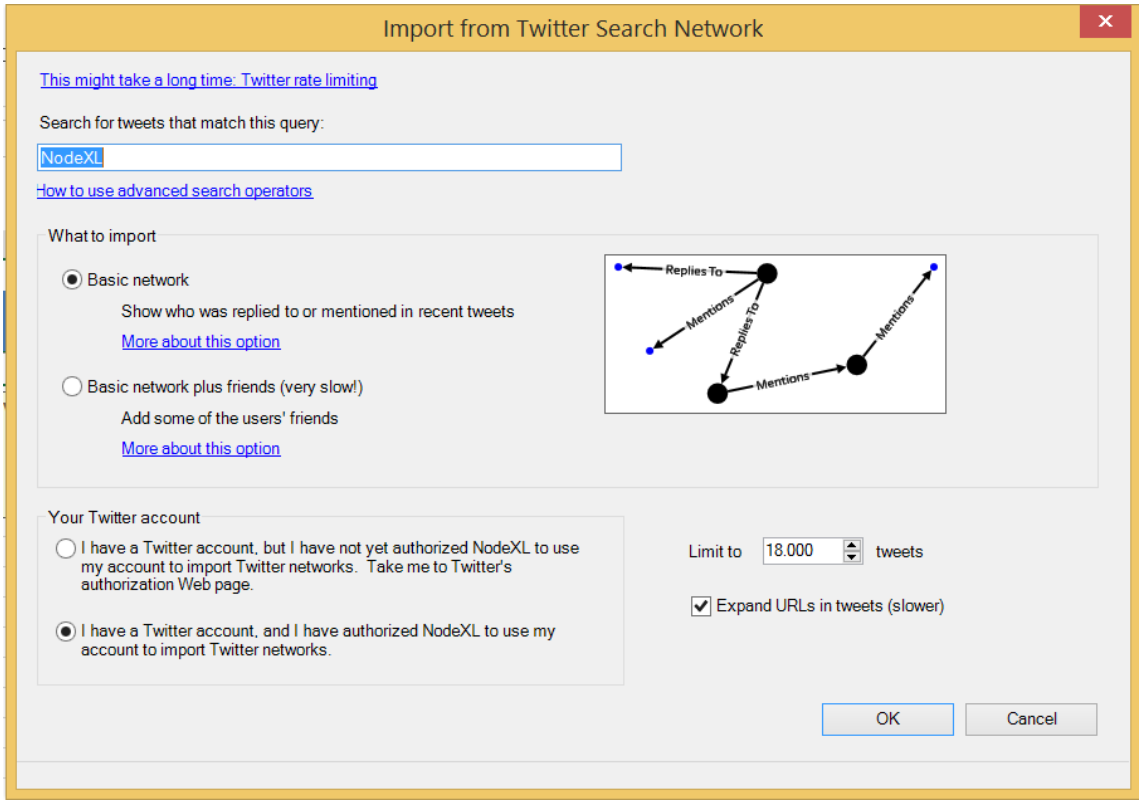
3.1.1.2 NodeXL

NodeXL és una aplicació molt semblant a TAGS, en aquest cas concret, aquesta aplicació és realment un add-in per Microsoft Excel.

A diferencia de TAGS, aquest afegit no només permet l'extracció de dades de la xarxa Twitter, sinó també d'altres xarxes, com Youtube o Flickr.

L'extracció de dades per Twitter, sent aquesta la principal font d'interès en aquest document, es pot realitzar mitjançant dos opcions:

- Extracció via Search API: Característica anàloga a la utilitzada per TABS. La imatge 6 mostra l'aspecte visual que ofereix NodeXL per aquesta opció.

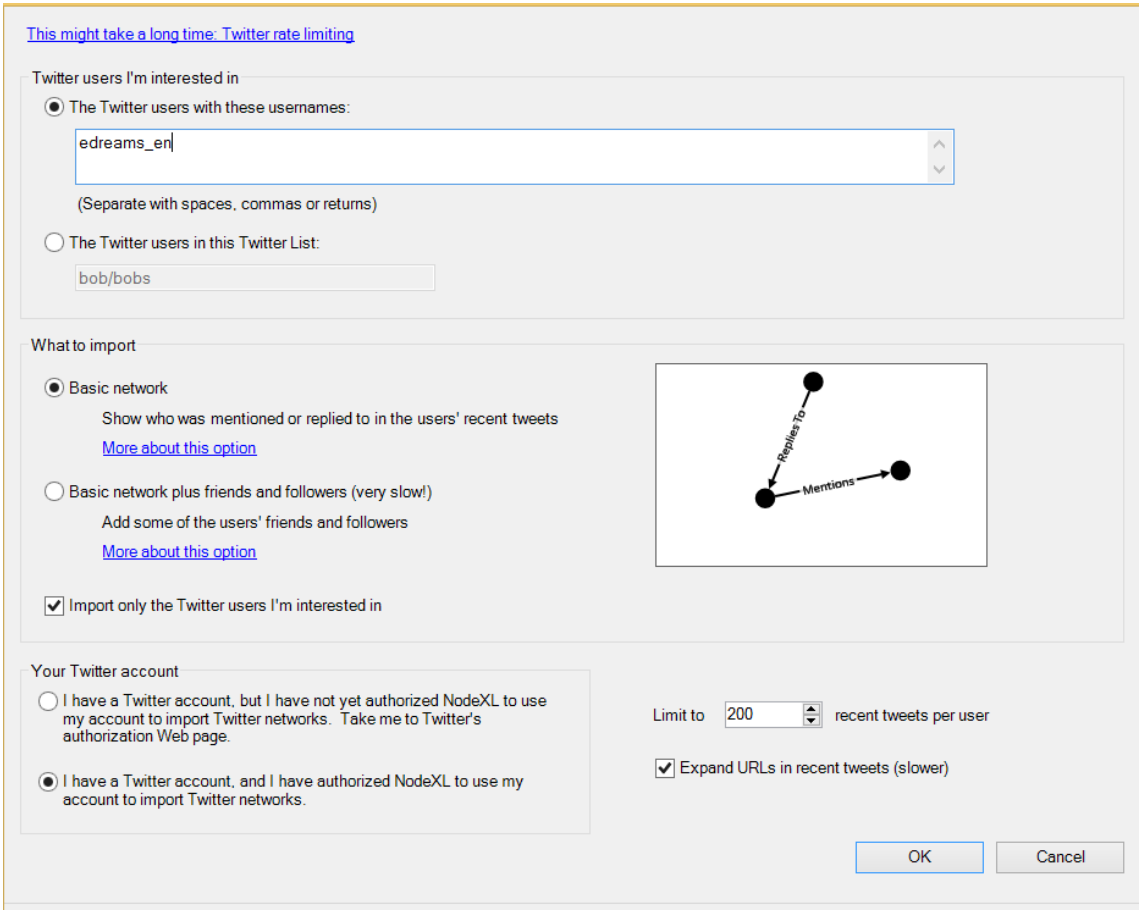


Imatge 6. Extracció mitjançant la Search API

- Extracció dels tweets de la pàgina d'un usuari: Com l'anterior, és una opció que ofereix de forma similar TAGS, la diferència principal consisteix en la possibilitat de realitzar cerques per més d'un usuari. La imatge 7 mostra la interfície per aquesta opció, es pot observar com ofereix la possibilitat de limitar els tweets per usuari o agafar els tweets d'usuaris que estan en la llista indicada.

Per qualsevol d'aquestes opcions, els resultats es mostren en la mateixa pàgina del llibre de càlcul, separant les dades per files i cada atribut d'un tweet en una columna diferent. Malauradament, les dades proporcionades són bastant més limitades que les obtingudes mitjançant TABS degut a que el llistat de dades generat és genèric sigui quina sigui la xarxa social. El lector podrà veure un exemple d'extracció d'aquestes dades en la imatge 5.

Finalment, NodeXL ofereix algunes opcions addicionals a la extracció, com la preparació de gràfiques, la importació de dades d'altres fonts, filtres dinàmics, etc... Aquestes opcions no són rellevants en la extracció de dades de les xarxes socials, així doncs, no s'hi entrarà en més profunditat.



Imatge 7. Extracció dels posts d'usuaris concrets

2	Vertex 1	Vertex 2	Relationshi	Relationship Date (UTC)	Tweet	URLs in Tweet	Domains in Tweet	Hashtags in Tweet	Tweet Date (UTC)	Twitter Page for Tweet	Latitude	Longitude	Imported ID	In-Reply-To Tweet ID
3	edreams_en	edreams_en	Tweet	##############	"At the end of the day, your feet sho			wanderlust tgif	##########	https://twitter.com/#!/edreams_en/statu			587276499012562945	
4	edreams_en	edreams_en	Tweet	13/04/2015 7:42	@KristieV	https://twi	twitter.com		##########	https://twitter.com/#!/edreams_en/statu			587521322177	587074919
5	edreams_en	edreams_en	Tweet	13/04/2015 7:43	@KristieVan_ Were you able to change the flight? If				##########	https://twitter.com/#!/edreams_en/statu			587521408969	587074919
6	edreams_en	edreams_en	Tweet	13/04/2015 7:43	@digitalt Hi, please send us a DM with your booking				##########	https://twitter.com/#!/edreams_en/statu			587521524602	586929123
7	edreams_en	edreams_en	Tweet	13/04/2015 9:55	Seven unk	http://ww	bbc.com		##########	https://twitter.com/#!/edreams_en/statu			587554617753530368	
8	edreams_en	edreams_en	Tweet	##############	Happy Tha	https://twi	twitter.com		##########	https://twitter.com/#!/edreams_en/statu			587561570756067329	
9	edreams_en	edreams_en	Tweet	##############	Travel tip	http://mag	edreams.com		##########	https://twitter.com/#!/edreams_en/statu			587565426541891585	
10	edreams_en	edreams_en	Tweet	##############	@KristieVan_ Sorry for the inconveniences. Can you				##########	https://twitter.com/#!/edreams_en/statu			587584187550	587565808
11	edreams_en	edreams_en	Tweet	##############	A mini gui	http://blog	edreams.com	traveltips	##########	https://twitter.com/#!/edreams_en/statu			587601646059786240	

Imatge 8. Resultats de la extracció de dades mitjançant NodeXL

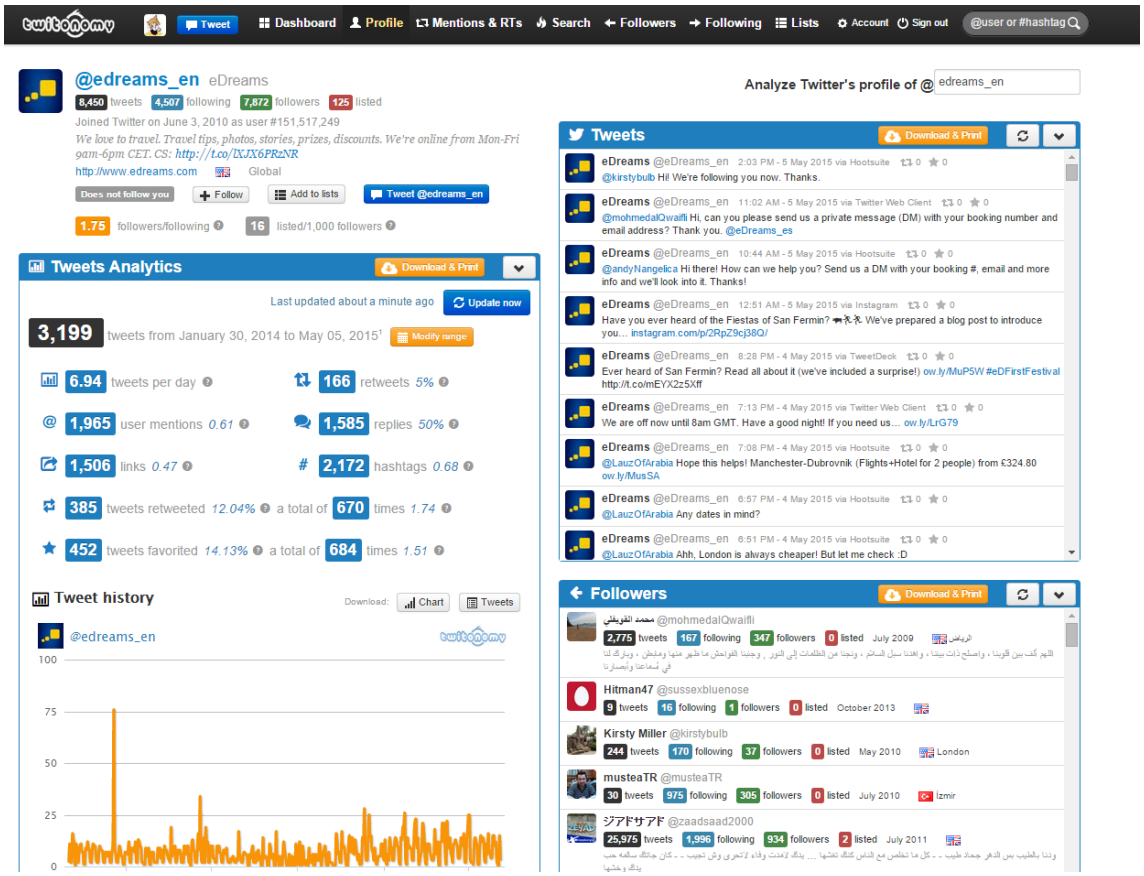
3.1.1.3 Twitonomy

Twitonomy, a diferència dels dos sistemes anteriors, és una aplicació web no dedicada específicament a la extracció de dades de Twitter, sinó més aviat a l'anàlisis d'aquestes dades. Aquesta aplicació és una de les moltes eines existents per tal de facilitar el dia a dia dels Community Managers de les pàgines de les marques.

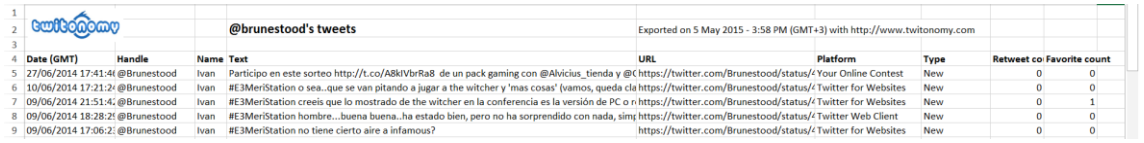
Entre moltes de les seves funcions, destaquen les estadístiques dels posts com: els tweets compartits, els dies de la setmana amb més activitat, les respostes rebudes, etc...

El motiu per la inclusió d'aquesta aplicació en l'apartat d'extracció de dades, és degut a que l'aplicació permet la descarrega dels tweets apareguts en una pàgina d'usuari en format Excel a un ordinador local.

La imatge 6 del document mostra la descarrega dels tweets del compte personal de l'autor d'aquest projecte. Com es pot observar, les dades descarregades són bastant escasses, oferint poc més que la data, el text, l'autor i el nombre de comparticions o favorits del tweet en qüestió.



Imatge 9. Pàgina inicial de Twitonomy



	twitonomy		@brunestood's tweets		Exported on 5 May 2015 - 3:58 PM (GMT+3) with http://www.twitonomy.com				
	Date (GMT)	Handle	Name	Text	URL	Platform	Type	Retweet co	Favorite count
	27/06/2014 17:41:4	@Brunestood	Ivan	Participo en este sorteo http://t.co/A8klVbrRa8 de un pack gaming con @Alvicius_tienda y @(	https://twitter.com/Brunestood/status/4	Your Online Contest	New	0	0
	10/06/2014 17:21:2	@Brunestood	Ivan	#E3MeriStation o sea..que se van pitando a jugar a the witcher y 'mas cosas' (vamos, queda cla	https://twitter.com/Brunestood/status/4	Twitter for Websites	New	0	0
	09/06/2014 21:51:4	@Brunestood	Ivan	#E3MeriStation creeis que lo mostrado de the witcher en la conferencia es la versión de PC o n	https://twitter.com/Brunestood/status/4	Twitter for Websites	New	0	1
	09/06/2014 18:28:2	@Brunestood	Ivan	#E3MeriStation hombre...buena buena..ha estado bien, pero no ha sorprendido con nada, simp	https://twitter.com/Brunestood/status/4	Twitter Web Client	New	0	0
	09/06/2014 17:06:2	@Brunestood	Ivan	#E3MeriStation no tiene cierto aire a infamous?	https://twitter.com/Brunestood/status/4	Twitter for Websites	New	0	0

Imatge 10. Resultats dels tweets descarregats mitjançant Twitonomy

3.1.2 Conclusions de l'anàlisis de les aplicacions actuals d'extracció de dades

Com el lector haurà pogut observar, el primer que li haurà sobtat és que en aquest anàlisis no ha aparegut cap aplicació per extreure dades de la xarxa Facebook, això és degut a que no es va trobar cap aplicació gratuïta per realitzar aquesta tasca en el moment de realitzar aquest estudi. El motiu, és possiblement la diferència en el tipus de privacitat aplicada en cada una de les xarxes. Mentre que Twitter, la majoria de les dades són públiques no passa així amb Facebook, on les dades per defecte no són públiques excepte en les pàgines de les marques.

Així doncs, al no poder trobar cap forma d'obtenir dades de forma automatitzada de la xarxa Facebook, provoca un problema en el cas de voler analitzar les seves dades.

Per altra banda, les aplicacions existents per extreure dades de Twitter, ofereixen poca versatilitat al haver de limitar-se únicament a les opcions existents, sense poder obtenir dades més concretes de forma còmode com els usuaris que han contestat un post, filtratge de resultats, etc...

Per aquests motius, es va decidir crear una aplicació des de cero amb la qual poder obtenir aquest tipus de dades d'ambdues xarxes sense més limitacions que les que ofereixen els propis serveis de Twitter i Facebook a l'hora d'oferir aquestes dades.

Abans de finalitzar aquest apartat, es presenta la taula 1 per tal de mostrar les diferències entre les aplicacions esmentades. Com es podrà observar, no s'entra en detall en les capacitats d'extracció per Facebook degut a que les aplicacions restants no disposen de cap funcionalitat envers aquesta xarxa social.

	TAGS	NodeXL	Twitonomy	Aplicació creada
Crides a l'API de Facebook	No	No	No	Si
Crides a la Search API(Twitter)	Si	Si	No	Si
Captura de la timeline(Twitter)	Si	Si	Si	Si
Exportació .csv	No	No	No	Si
Configuració dels tweets de sortida	No	No	No	Si
Realització de gràfiques	No	No	Si	No
Consultar respostes d'un tweet/post	No	No	No	Si

Taula 1. Diferències entre les aplicacions analitzades

En els següents apartats s'explicarà el funcionament bàsic dels serveix que ofereixen tant Twitter i Facebook per oferir les dades existents en les seves xarxes.

3.1.3 Característiques comunes dels serveis de captura de dades

Abans de procedir a descriure les característiques pròpies de cadascuna de les APIs, és necessari explicar breument alguns conceptes que comparteixen ambdós xarxes per tal de proporcionar al lector una base per entendre correctament els serveis que ofereixen aquestes per tal de recol·lectar dades.

Les dues xarxes socials ofereixen les seves dades a través d'una API, acrònim de *Application Programming Interface*, per tal de que qualsevol persona amb coneixements de programació pugui accedir amb les seves pròpies aplicacions a aquestes xarxes.

Aquestes APIs, Graph API en cas de Facebook i REST i Streaming API en el cas de Twitter, proporcionen als usuaris una sèrie de direccions web, anomenats *endpoints*, mitjançant els quals un cop hi accedeix un usuari, la xarxa li retorna una sèrie de dades segons l'endpoint que hagi accedit, sempre en el cas de que l'usuari tingui autorització per accedir a aquesta informació. Si l'usuari accedeix a l'endpoint que retorna els posts de la pàgina principal d'una marca, aquesta serà la informació que li proporcionarà l'API sempre i quan l'usuari tingui autorització.

Les xarxes de Twitter i Facebook retornen la informació en un format concret anomenat JSON, acrònim de JavaScript Object Notation, un tipus de notació que deriva tal com diu el nom, del llenguatge de programació JavaScript. La imatge 11 mostra un exemple de text en format JSON extret directament d'una crida a un endpoint de l'API de Facebook on es poden veure clarament les dades traspassades a l'usuari amb molta facilitat.

Aquest format no és només utilitzat per Twitter i Facebook, sinó que s'ha convertit en un estàndard de transferència de dades utilitzat en molts àmbits, des de les xarxes ja esmentades, a comunicacions d'un sistema Android a algun servei web.


```
{
 "id": "10152641144469273",
 "first_name": "Ivan",
 "gender": "male",
 "last_name": "Bargalló“,
 "locale": "es_ES",
 "name": "Ivan Bargalló",
 "timezone": 1,
 "updated_time": "2014-11-03T00:30:24+0000",
 "verified": true
}
```


Imatge 11. Exemple de text en format JSON

Finalment, la imatge 12 mostra en un exemple bàsic a mode de resum, els passos a seguir per tal d'obtenir dades de les xarxes socials a través de les seves APIs.

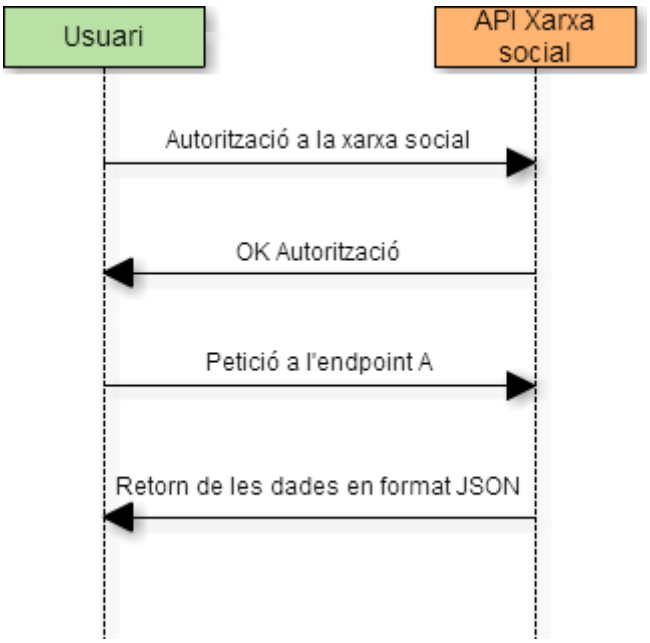
Un cop mostrades les característiques comunes, els subapartats següents descriuran al lector les característiques pròpies de cadascuna de les APIs de les xarxes.

3.1.3.1 API de Twitter

3.1.3.1.1 Característiques generals

Per tal d'extreure informació sobre la xarxa social, Twitter ofereix dos APIs, la REST API i la Streaming API, cadascuna amb diferents capacitats per tal d'ajudar als desenvolupadors a aconseguir la informació que els hi sigui necessària.

Figura 1.



Imatge 12. Passos a seguir per obtenir dades de les xarxes socials. Font: Elaboració pròpia

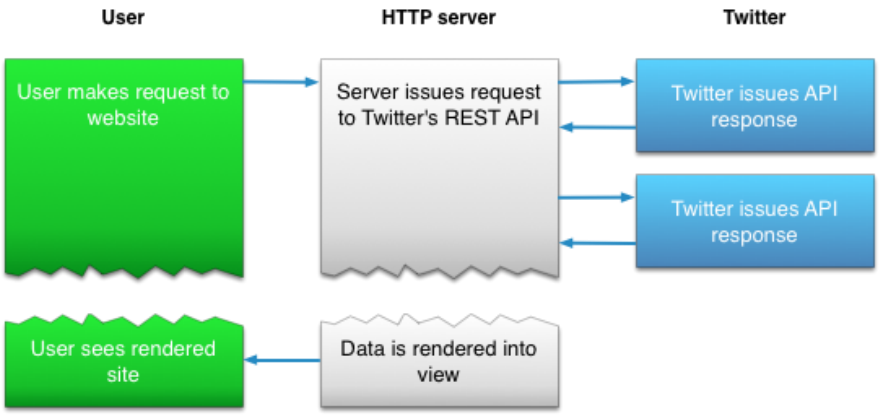
Sigui quina sigui l'API escollida a la qual realitzar les crides, Twitter proporciona sempre les dades de retorn en forma d'algun dels següents quatre objectes:

- **Tweet**: Entitat bàsica de Twitter que representa un tweet tal com diu el seu nom. Retorna entre altres camps el text del tweet, la data, el nombre de retweets, etc...
- **User**: Tot tweet té un autor, aquest objecte proporciona tota la informació pública disponible del seu autor. Alguns dels seus camps són la data de creació, el nom de l'autor, l'URL de la seva foto de perfil, etc...
- **Entity**: Les entitats són l'objecte més abstracte dels quatre, aquestes representen diversos objectes amb pocs camps que representen algun tipus d'informació addicional que l'usuari ha afegit al seu tweet. Tan pot ser un objecte de tipus URL, de la qual disposarem de l'enllaç original, la versió acurtada, etc.. com un de tipus MEDIA, del qual podrem obtenir el tipus , és a dir, si és per exemple una foto, l'enllaç original, la mida, entre altres.
- **Place**: L'objecte Place, tal com es pot suposar, retorna a l'usuari informació sobre la localització on habitualment s'ha escrit el tweet. Les coordenades, el país, etc... són alguns dels camps que es poden trobar en aquest objecte.

A continuació es descriuran les possibilitats de les dos APIs existents a Twitter i les seves diferències fonamentals.

3.1.3.1.2 REST API

L'API REST de Twitter és una API de tipus crida/resposta, és a dir, cada cop que l'usuari realitza una crida a l'endpoint en qüestió, l'API retorna la informació demanada. La imatge 13 mostra el seu funcionament.



Imatge 13. Funcionament de la REST API. Font: Twitter

La informació que pot proporcionar l'API és molt variada, a continuació se'n proporcionen alguns exemples destacats:

- Cercar tweets que tenen una sèrie de característiques mitjançant la Search API, una API anteriorment separada de la REST API però que amb la versió actual s'ha fusionat convertint-se en un endpoint d'aquesta última.
- Obtenir la pàgina d'inici o *timeline* d'un usuari.
- Llegir o escriure els missatges privats d'aquest.
- Escriure estats de Twitter en un compte d'usuari.
- Obtenir les respostes d'un tweet.

L'accés a aquestes funcions, no és il·limitat, Twitter imposa una sèrie de limitacions temporals als endpoints de la REST API per tal d'evitar una sobresaturació i abús dels seus servidors. D'aquesta manera, per tal de facilitar la limitació, la xarxa social ofereix dos tipus d'accés:

- Accés per usuari: L'identificador d'accés, o *token*, pertany a un usuari en concret, aquest podria ser el cas de per exemple en que aquest hagués permès a una aplicació que utilitzi Twitter (un client) en el seu nom, així doncs, totes les peticions es realitzarien en el nom del client.
- Accés per aplicació: En aquest cas, el token d'accés pertany a l'aplicació, així, no seria necessari que els usuaris que utilitzessin l'aplicació proporcionessin cap tipus de permís a aquesta per tal d'utilitzar-la, doncs totes les peticions a l'API es realitzaran en nom de l'aplicació.

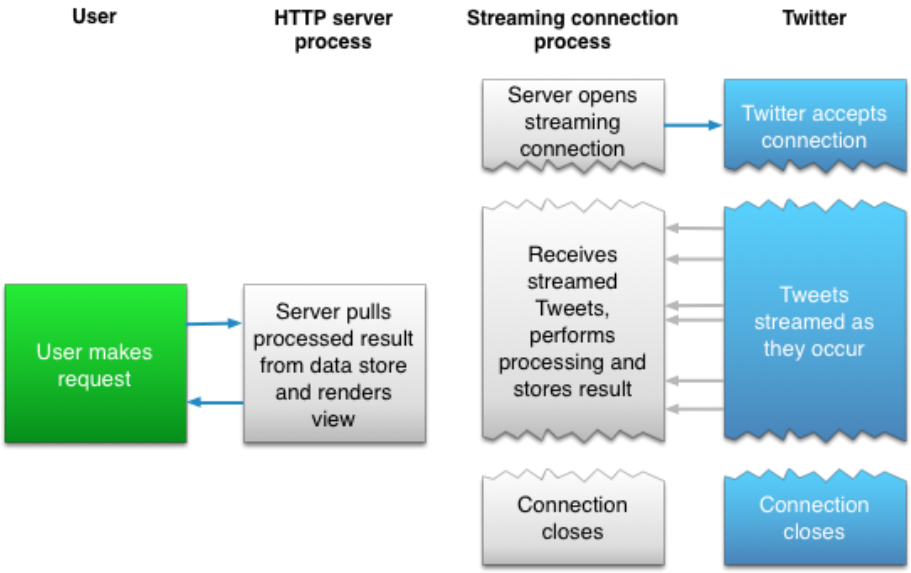
Un cop definits els dos tipus d'accés, Twitter, en la nova versió de l'API, la xarxa limita els nombre d'accessos a la seva API en blocs de 15 minuts, és a dir, en el moment en que es passa d'un bloc de 15 minuts a un altre, Twitter reinicia el comptador d'accessos a la seva API. Així per exemple, en el cas de que es vulgui utilitzar la Search API per buscar tweets amb un hashtag concret, tenim una limitació de 180 accessos d'usuari i 450 d'aplicació, o bé, de 180 i 300 respectivament, en el cas de voler obtenir els tweets de la pàgina d'inici, o *timeline,* d'un usuari en concret.

Finalment, els endpoints i les restriccions nombrades només són algun dels molts casos existents en l'API REST.

3.1.3.1.3 Streaming API

La Streaming API de Twitter, a diferència de l'API REST, té el propòsit de aconseguir dades de la xarxa social en temps real, no ofereix fer cerques en el passat tal com succeeix amb aquesta última, sinó que s'obtindran únicament les noves dades que apareguin des de el moment que es crida l'endpoint.

El funcionament tècnic consisteix en realitzar una connexió permanent amb el servidor de Twitter, un procés intermedi capturarà les noves dades que va enviant aquest segons va apareixent més informació a la xarxa social filtrant segons sigui necessari, per al final retornar-la a l'usuari quan aquest realitza una petició a aquest procés. La imatge 14 mostra l'esquema de funcionament d'aquesta API que ofereix Twitter als seus usuaris:



Imatge 14. Diagrama de funcionament de la Streaming API. Font: Twitter

En el cas d'aquesta API, Twitter ofereix tres tipus de streaming amb el seus propis endpoints per tal de recaptar les dades que siguin necessàries, aquests són: els public streams, els user streams i el site streams.

Els següents subapartats donaran una breu descripció sobre els streams esmentats.

3.1.3.1.3.1 Public stream

El Public stream possiblement sigui el més utilitzat per la gent que pràctica mineria de dades, aquest tipus de stream captura dades mitjançant tres mètodes:

- **Filtratge de tweets**: L'endpoint de filtratge de tweets pot capturar dades en temps reals bé sigui d'usuaris concrets, en localitzacions concretes o bé tweets que continguin una sèrie de paraules clau. Com a limitació important, no s'obtindran tots els tweets que compleixin el filtre si el total d'aquests suposa més d'un 1% del total de tweets enviats a la xarxa social.
- **Mostra de tweets**: El segon endpoint permet aconseguir una mostra de tots els tweets que s'envien a la xarxa social sense aplicar cap tipus de filtrat.

- **Firehose**: L'últim mètode és possiblement la font de dades més potent que pot oferir Twitter. Aquest endpoint, únicament accessible mitjançant un acord directe amb Twitter o algun dels seus proveïdors autoritzats, és l'única alternativa existent per aconseguir tots i cadascun dels estats públics de la xarxa.

3.1.3.1.3.2 User Stream

Aquest stream conté un únic mètode per capturar dades. La funció d'aquest és retornar tots els tweets en temps real de l'usuari autenticat.

3.1.3.1.3.3 Site streams

L'últim stream, de recent creació, encara està en fase beta. El seu únic endpoint permet recuperar les dades dels usuaris que han autoritzat a una aplicació recollir les seves dades en temps real, sent aquesta aplicació l'usuari que realitza la connexió amb Twitter. A mode d'exemple, aquest servei permetria que una web amb compte de Twitter, mostres timelines de diversos usuaris a la vegada sempre que aquests haguessin donat la corresponent autorització.

3.1.3.2 Api de Facebook

3.1.3.2.1 Característiques bàsiques

Facebook, a diferència de Twitter només ofereix una sol'API mitjançant la qual es poden extreure dades, la Graph API. La navegació per aquesta API, tal com indica el seu nom, segueix la forma d'un graf, és a dir, navegant des d'un node fins un altre a partir d'un vèrtex concret, s'aconseguirà nova informació. En aquest cas, els elements bàsics de la Graph API, com els de qualsevol graf són els següents:

- Nodes: Objectes de Facebook, com per exemple usuaris, fotos, comentaris, etc... Cadascun d'ells amb un seguit d'atributs per definir l'objecte, com el nom de l'usuari pels objectes d'usuari, el text o la data de creació d'un post en cas de l'objecte post, etc..
- Vèrtex: Les connexions entre aquests objectes. Per exemple, existeix un vèrtex que connecta un post a la xarxa social amb tots els usuaris que han pitjat *like* sobre aquest.

Al igual que l'API de Twitter, la Graph API funciona amb el mateix sistema mitjançant endpoints, és a dir, un cop autoritzat l'usuari, es realitza una petició a una adreça definida per l'API, oferint informació variable en format JSON segons l'adreça a la qual s'ha realitzat la petició.

Addicionalment, al igual que l'API de Twitter també es permet escriure nova informació a la xarxa mitjançant l'API, o fins i tot esborrar-la, però degut a que l'objectiu del projecte només contempla l'opció de extreure informació, en aquest document no es proporcionarà més informació dels endpoints corresponents a aquestes tasques.

3.1.3.2.2 Peticions imbricades

Una característica diferencial de la Graph API de Facebook, és la capacitat per imbricar diverses peticions en una sola, per exemple, la imatge 15 mostra com en una sola petició a un

endpoint de l'API es poden capturar de cinc àlbums d'un usuari, les dos primeres fotos, a més del nom, l'URL de la fotografia i la gent etiquetada d'aquesta última.

```
GET graph.facebook.com/me?fields=albums.limit(5).fields(name,
photos.limit(2).fields(name, picture, tags.limit(2)))
```

Imatge 15. Exemple de peticions imbricades

3.1.3.2.3 Public Feed API

La Public Feed API és una API anàloga a la Streaming API, és a dir, és una API mitjançant la qual es poden obtenir dades en temps real. La diferència principal amb aquesta consisteix en que la seva utilització està limitada a usuaris que han arribat a un acord econòmic amb Facebook, motiu pel qual queda completament fora de l'abast del projecte.

3.1.3.2.4 Cerca de posts

La cerca en Facebook es extremadament limitada, tot i existir un endpoint per tal de fer cerques a la xarxa social, aquest endpoint està limitat a cercar usuaris, pàgines, events, grups, llocs o localitzacions, sent aquests definits com a llocs que s'han creat per l'usuari o amics de l'usuari realitzador de la cerca, o llocs els quals l'usuari ha sigut etiquetat.

Anàlogament a Twitter, Facebook ofereix una opció de cerca addicional similar a Firehose, anomenada Keyword Insights, en la qual la potència de la cerca és molt major. La problemàtica és similar a la de l'apartat anterior, l'accés a Keyword Insights està limitada a aquells usuaris amb permisos especials atorgats per Facebook.

3.1.3.2.5 Restriccions d'accés

A diferència de Twitter, on indica clarament les limitacions existents a la hora de realitzar crides a les seves APIs, aquesta informació no està disponible a la documentació de Facebook.

3.2 Machine learning

En els següents apartats s'introduirà l'àrea de Machine Learning del projecte, es donarà una petita introducció sobre aquest, les seves categories, la part en la que s'ha centrat el projecte i una breu explicació sobre els algoritmes utilitzats en aquest.

3.2.1 Introducció

Arthur Samuel, va definir al 1959 el Machine Learning com "el camp d'estudi que dona a un ordinador la capacitat d'aprendre sense estar específicament programat" (Simon et al., 2013), és a dir, el camp que estudia les possibilitats que té un ordinador en aprendre a realitzar tasques sobre les quals no ha sigut instruït directament.

Tot i que l'àrea del Machine Learning sempre ha tingut una evolució constant, creant nous algoritmes o redactant estudis per analitzar l'eficiència d'aquests, aquests últims anys gràcies a l'apogeu de les xarxes socials i el *Big Data*, l'interès per la matèria ha augmentat de forma bastant important. Aquest interès provocat per la relació apareguda entre les xarxes socials i el Machine Learning ha sigut una de les motivacions per la realització d'aquest projecte.

L'estudi de la utilització d'algoritmes d'aprenentatge automàtic a les xarxes no és una novetat inclosa en aquest projecte, tot al contrari, una subcategoria d'aquesta àrea coneguda com *Sentiment Analysis,* ha sobrepassat l'àmbit acadèmic per convertir-se en un factor a tenir en compte per les empreses que vulguin establir una política de comunicació a la xarxa. En un apartat posterior del document, es proporcionaran més detalls sobre aquesta àrea degut a la relació que pot tenir aquesta en el projecte.

L'aparició de noves tècniques i algoritmes ha permès dividir l'àrea del Machine Learning en una sèrie de branques o categories diferenciant-se principalment entre elles per la forma en que els algoritmes són entrenats. En el punt següent es proporcionarà informació sobre les diferents categories existents.

3.2.2 Categories de Machine Learning

Tal com s'ha esmentat en l'apartat anterior, existeixen diversos tipus d'aprenentatge utilitzat per educar als algoritmes. Els següents punts explicaran aquests tipus, centrant-se principalment en l'aprenentatge supervisat, degut al ser l'utilitzat en aquest projecte.

3.2.2.1 Aprenentatge supervisat

3.2.2.1.1 Introducció

L'aprenentatge supervisat consisteix en fer prediccions a futur basant-se en comportaments o característiques que l'algoritme ja ha observat en dades ja estudiades. Per exemple, proporcionant una sèrie d'informació de correus els quals ja estan classificats prèviament per *spam* i no spam, un algoritme podria observar que tots els correus de spam provenen de certa adreça IP, o bé d'una direcció de correu amb un domini concret. Amb aquesta informació disponible l'algorisme podria deduir que els nous correus rebuts que compleixin les mateixes característiques siguin classificats de la mateixa forma.

Les característiques utilitzades pels algoritmes, poden ser clarament definides com en l'exemple anterior, com pot ser l'IP d'origen, o bé definides de forma lleugerament més

abstracte, en l'àmbit del projecte concretament, les frases dels datasets proporcionats han de ser transformades en vectors numèrics els quals, cada posició d'aquest vector indica informació relativa de la freqüència d'aparició d'una paraula en el conjunt total del data set.

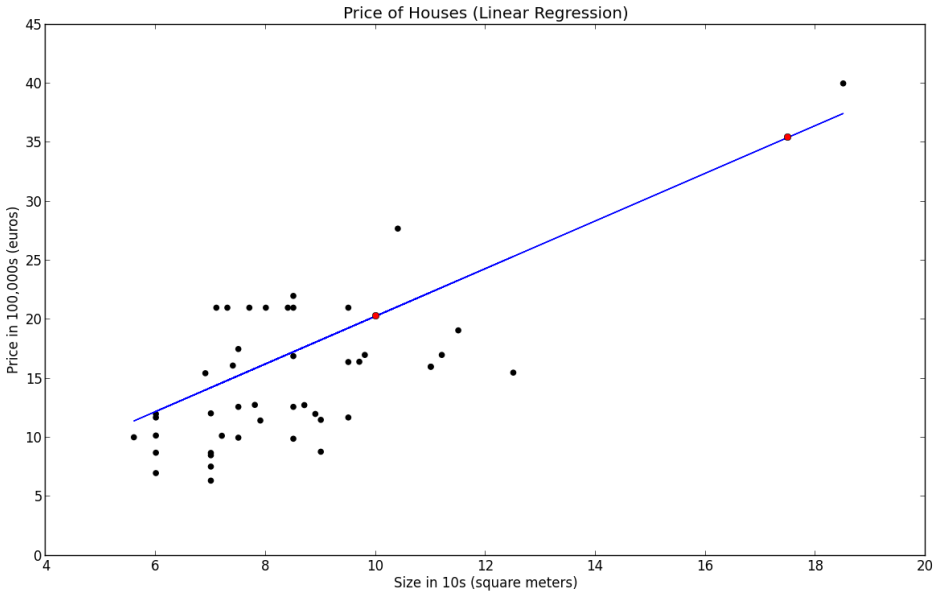
L'aprenentatge supervisat, permet resoldre dos classes de problemes principalment, els problemes de regressió i els problemes de classificació.

3.2.2.1.2 Problemes de regressió

Els problemes de regressió són aquell tipus de problemes els quals es basen en la predicció d'un valor continu, habitualment un nombre real, mitjançant un conjunt de característiques ja conegudes.

Un dels exemples més clàssics de problemes de regressió que es podria resoldre amb algoritmes de Machine Learning, seria la predicció del preu de venta d'un habitatge coneixent el nombre d'habitacions, els seus metres quadrats o la seva orientació, entre altres.

La gràfica 1, extreta del curs de Machine Learning de Stanford, mostra l'exemple anterior de forma gràfica, en aquest cas, es poden veure en forma de punts negres tots els valors amb els quals s'ha entrenat l'algoritme, mentre que els vermells són les prediccions que es volen realitzar. El segment blau indica el model que s'ha creat a partir de l'algoritme de Machine Learning, gràcies al qual cada cop que es vulgui fer una nova predicció, s'haurà de cercar la intersecció entre el segment i els metres quadrats per conèixer la predicció resultant del model.



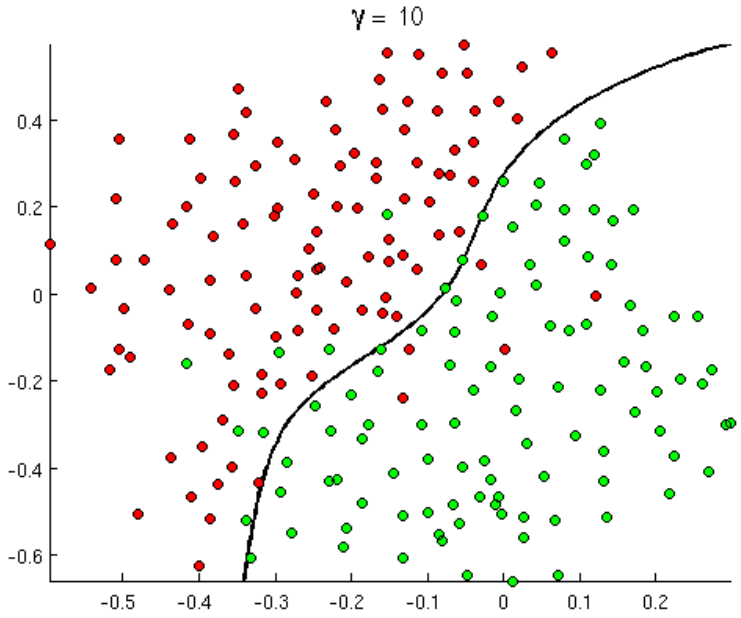
Gràfica 1. Representació d'un problema de regressió. Font: Curs ML Stanford

3.2.2.1.3 Problemes de classificació

Els problemes de classificació són aquells que busquen classificar un conjunt d'instàncies a unes classes concretes ja definides. A continuació, s'esmenten alguns problemes que es poden resoldre utilitzant algoritmes de classificació.

- Utilitzar les característiques d'un tumor per saber si és benigne o maligne.
- Classificar una sèrie de correus per spam o no spam.
- Categorització de textos en temàtiques.
- La idoneïtat de concedir una hipotètica a un client comparant les seves dades amb les hipoteques concedides a clients anteriors amb característiques similars.

Per proporcionar al lector una explicació més visual sobre els problemes de classificació, la gràfica 2, extreta de la mateixa curs de Stanford ja mencionat anteriorment, mostra una representació gràfica del model obtingut amb un algoritme de classificació i les instàncies utilitzades pel seu entrenament. En aquest gràfic, les instàncies estan classificades en dues classes, representades cadascuna en un punt d'un color diferent mentre que la corba del gràfic és el llindar que limita la pertinència a una classe que proporciona el model creat per l'algoritme, així doncs, cada cop que s'introdueix una instància nova per a que sigui classificada al model, depenent de la part de la gràfica on estigui situada aquesta nova instància estarà classificada en una classe o en un altre.



Gràfica 2. Representació d'un problema de classificació. Font: Curs de ML de Stanford

Finalment, la categorització de textos ha sigut el tipus de classificació utilitzat durant la realització d'aquest projecte, motiu pel qual en l'apartat 3.2.3 d'aquest document es pot veure una descripció sobre aquesta. Tot i així, per aquest motiu, mentre que en aquest punt es proporciona una definició general sobre els problemes de classificació, en un apartat posterior es proporcionaran més detalls sobre ella.

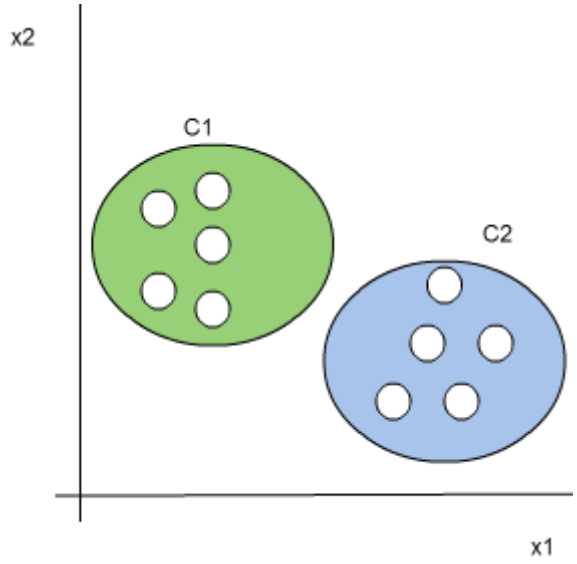
3.2.2.2 Aprenentatge no supervisat

3.2.2.2.1 Introducció

L'aprenentatge no supervisat a diferència del supervisat consisteix en extreure conclusions de datasets formats per una sèrie de dades sense classificar de forma inicial. Per aquest motiu, com no existeix cap tipus de categorització prèvia, no hi ha cap tipus de senyal per avaluar una solució potencial.

3.2.2.2.2 Clustering

L'aprenentatge no supervisat està compost per una varietat de mètodes per inferir informació d'utilitat d'un conjunt de dades d'entrenament. Un dels mètodes principals és l'anomenat *clustering*, el qual consisteix en agrupar les dades a partir de la seva semblança mitjançant clústers.



Gràfica 3. Representació de dos clústers en un espai bidimensional. Font: Elaboració pròpia

Aquesta tècnica té una gran quantitat d'aplicacions reals avui en dia, com per exemple en l'aplicació de Google, Google News. L'empresa cerca cada dia entre els centenars de milers de noticies existents en el dia a dia de la societat, aplica un algorisme de clustering a elles obtenint com a resultat, un seguit de clústers o temes comuns amb els quals pot organitzar les noticies al seu portal. Altres exemples poden trobar-se a l'hora de realitzar investigacions de mercat, gràcies al clustering es poden trobar característiques comunes en un dataset de clients d'una empresa, gràcies a les quals es podran utilitzar tècniques personalitzades de venta al segment trobat.

Finalment, durant la primera etapa del projecte en la que es va realitzar una primera investigació sobre el Machine Learning, es va poder observar que tot i que l'objectiu del projecte era utilitzar aprenentatge supervisat per poder classificar textos, la possibilitat d'utilitzar aprenentatge no supervisat en la classificació de textos era una idea raonable. Aquest fet es va observar en els resultats positius obtinguts en estudis on s'utilitzaven algorismes de clustering per classificar les opinions d'usuaris escrites en una revista electrònica (Turney et al., 2002).

3.2.3 Classificació de textos

En els següents apartats és proporcionarà al lector una breu descripció de la classificació de textos per tal d'introduir-lo a la matèria.

3.2.3.1 Introducció

La classificació de textos és la branca del Machine Learning que s'encarrega de proporcionar una categoria a una instància de text.

Aquesta branca, al ser un subconjunt de l'àmbit de classificació d'objectes en l'aprenentatge màquina, tot i utilitzar les mateixes eines d'aquest àmbit per resoldre els seus problemes, té algunes particularitats necessàries a tenir en compte a l'hora de realitzar qualsevol tipus d'experimentació.

3.2.3.2 Característiques de la classificació de textos

El funcionament de la classificació de textos té una sèrie de característiques importants a tenir en compte. Mentre que en els problemes de classificació més generals els atributs poden ser utilitzats directament en els diversos algoritmes, és necessari utilitzar una sèrie de processos previs en el conjunt de dades d'entrada per tal poder realitzar la classificació de textos.

Alguns dels processos es detallen a continuació.

3.2.3.2.1 Bag of words

El primer procés a comentar és la bossa de paraules. Aquest mètode crea un vector on cada posició representa una paraula apareguda en el corpus. A continuació, cada instància representada en aquest, és convertirà en un nou vector de tantes posicions com paraules existeixen en el corpus. Finalment, cada posició d'aquest vector recent creat contindrà el nombre d'ocurrències de la paraula representada en la seva posició en la instància. A continuació es mostra un exemple del seu funcionament.

Tenint a la taula 2 el següent conjunt d'instàncies:

Instància	Text
1	I loved the movie
2	The book was horrible

Taula 2. Corpus d'exemple per la bossa de paraules

Utilitzant el model de bossa de paraules explicat, es pot extreure el següent vector:

Paraules úniques
<I, loved, the, movie, The, book, was, horrible>

Taula 3. Vector de paraules resultant

Un cop obtingut el vector, només cal transformar les instàncies inicials del corpus en un vector que representa l'aparició de les paraules corresponents tal com es pot veure a la taula 4.

Instància	I	Loved	The	Movie	The	Book	Was	Horrible
1	1	1	1	1	0	0	0	0
2	0	0	0	0	1	1	1	1

Taula 4. Representació de les noves instàncies

3.2.3.2.2 Transformació TF-IDF

En l'apartat anterior es mostra com cada element del vector generat per la bossa de paraules representa la freqüència d'aparició d'una paraula en una instància. La transformació TF-IDF és una alternativa a la utilització de la freqüència com a valor del vector.

Definida com *Term frequency – Inverse document frequency,*la transformació TF-IDF és una mesura utilitzada per indicar la rellevància d'una paraula en una col·lecció d'instàncies.

La freqüència del terme té diversos mètodes de càlcul.

- Freqüència bruta: Es calcula únicament el nombre d'aparicions del terme en una instància. Es pot denotar amb la formula 1.

$$tf(f,d) = f(t,d)$$

Formula 1. Freqüència bruta

- Freqüència booleana: La freqüència booleana és utilitzada per indicar si el terme apareix o no en la instància corresponent.

$$tf(t,d) = 1\ si\ t\ apareix\ en\ d, 0\ en\ cas\ contrari$$

Formula 2. Freqüència booleana

- Freqüència escalada logarítmicament: En aquest càlcul es té en compte el logaritme de la freqüència bruta.

$$tf(t,d) = 1 + \log f(t,d)\ \text{ si f(t, d)} > 0, 0\ en\ cas\ contrari$$

Formula 3. Freqüència escalada logarítmicament

- Freqüència normalitzada: Per tal d'evitar instàncies massa llargues, s'utilitza aquest tipus de càlcul que limita el valor amb la freqüència bruta màxima de les paraules de la instància.

$$tf(t,d) = \frac{f(t,d)}{\max\{f(w,d) : w\ \in\ d\ \}}$$

Formula 4. Freqüència normalitzada

La inversa de la freqüència és una mesura que denota si el terme és comú o no en el dataset complert. La formula 5 mostra el càlcul utilitzat.

$$idf(t,D) = \log \frac{|D|}{|\{d\ \in\ D : t\ \in\ d\}|}$$

Formula 5. Càlcul de la idf

On:

- |D| és el nombre d'instàncies del dataset.
- $|\{d\ \in\ D : t\ \in\ d\}|$ és el nombre d'instàncies on apareix el terme t. En cas de no existir es dividirà per 1.

Finalment, amb aquests dos valors, la transformació TF-IDF és calcula mitjançant la formula 6.

$$tfidf(t,d,D) = tf(t,d) * idf(t,D)$$

Formula 6. Càlcul de tf-idf

3.2.3.2.3 Selecció d'atributs

Mentre que els processos anteriors són propis per la resolució dels problemes de classificació de textos, el procés de selecció d'atributs, un dels processos més importants a realitzar, s'utilitza també en els problemes generals de classificació.

La selecció d'atributs es defineix com el procés de seleccionar un subconjunt d'atributs del dataset d'entrenament per ser utilitzat com a característiques de la categorització de textos.

Aquest procés persegueix dos objectius:

- **Disminuir el volum d'atributs**: El volum d'atributs que es poden generar mitjançant els mètodes anteriors és un nombre massa elevat com per poder ser tractat amb eficiència, per aquest motiu, el procés de selecció d'atributs pot arribar a ser necessari per tal de disminuir el nombre d'aquest en els datasets d'entrada.
- **Augmentar la precisió dels algoritmes**: A aquella característica que quan es afegida en el dataset augmenta l'error de classificació de noves dades se l'anomena soroll. El procés de selecció d'atributs pot ajudar a eliminar aquest tipus de característiques menys importants.

3.2.3.3 Aplicacions de la classificació de textos

La categorització de textos poden ser aplicades a molts camps diferents, a continuació se'n veuen alguns exemples:

- Filtre de spam.
- Classificació de documents.
- Identificació de llenguatges.
- Categorització de noticies.
- Sentiment analysis.

L'anàlisi de sentiments és una de les motivacions per dur a terme aquest projecte, per aquest motiu en el següent apartat s'oferirà informació més detallada sobre aquesta matèria.

3.2.4 Sentiment analysis

En el següent apartat es mostrarà al lector una explicació sobre l'anàlisi de sentiments, en la qual s'explicarà que és, la seva importància en la societat actual i quina relació té amb la classificació de textos.

3.2.4.1 Introducció

L'anàlisi de sentiments o sentiment analysis, és el camp d'estudi que analitza la opinió de la gent, els sentiments, avaluacions, actituds i emocions cap a una entitat com pot ser un producte, servei, empreses, individus, problemes, esdeveniments i matèries (Liu et al., 2012).

Existeixen molts termes amb els quals l'anàlisi de sentiments pot ser relacionat: mineria d'opinió, extracció d'opinió, mineria de sentiments, anàlisi de subjectivitat o anàlisi d'emoció, són només alguns dels molts noms existents.

Tot i que el terme va ser emprat per primera vegada per identificar la forma en que els sentiments són expressats en textos (Nasukawa et al., 2003), en anys anterior ja es van començar a realitzar recerques sobre opinions i sentiments, sent un dels pioners l'article de Turney sobre la matèria.

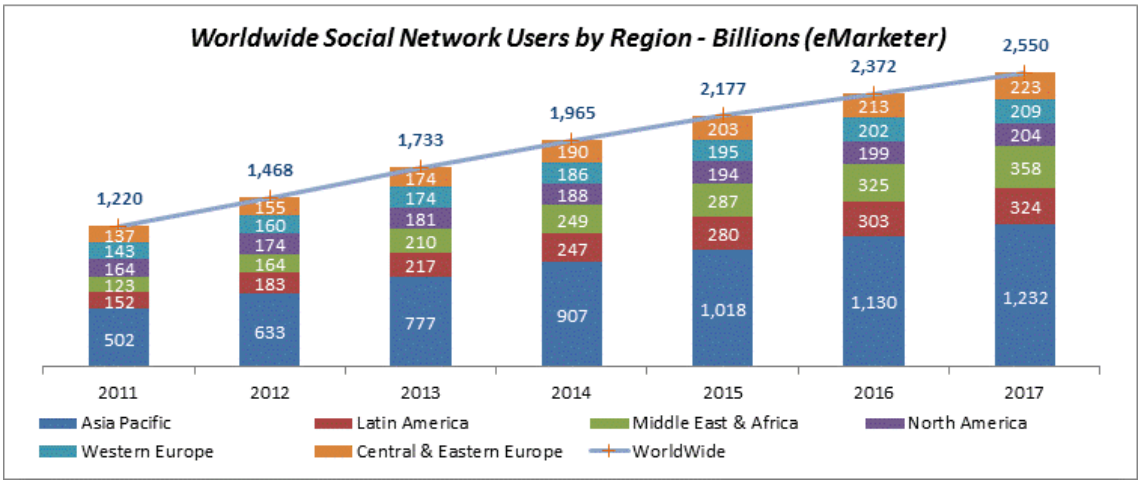
3.2.4.2 Importància del Sentiment analysis

Saber el que pensa la gent sempre ha sigut un dels factors principals utilitzats per la societat a l'hora de prendre una decisió. En qualsevol moment a l'hora de realitzar una compra, fer un

viatge o consumir un servei, utilitzar el coneixement de la gent més pròxima és un dels passos a realitzar per tal d'obtenir la major informació possible.

Anteriorment, les empreses per tal de conèixer la opinió dels seus consumidors realitzaven enquestes o formaven grups de discussió, convertint-se aquest tipus de tasques en una de les principals feines realitzades pels departaments de màrqueting.

Avui en dia, amb l'aparició d'internet i les xarxes socials, el nombre de gent a la qual es pot consultar la opinió, ha crescut de forma exponencial paral·lelament amb la quantitat d'usuaris d'aquestes. Per aquest motiu, al desaparèixer la limitació d'obtenció d'informació, es fa pràcticament impensable realitzar qualsevol compra de productes o serveis sense consultar la xarxa amb antelació.



Imatge 16. Estimació del nombre d'usuaris a les xarxes social. Font: eMarketer

Per les organitzacions, el paradigma també ha canviat. La necessitat de la creació d'enquestes i grups de discussió ha disminuït segons anava augmentat la possibilitat d'obtenir les dades dels clients directament de la xarxa. Tot i així, aquest augment d'informació ha generat dos inconvenients importants que una organització ha de solucionar:

- Proliferació dels llocs d'opinió: les empreses poden arribar a tenir problemes per obtenir les dades més importants per tal de realitzar l'anàlisi més adient. Per exemple, un restaurant no només hauria de ser capaç d'analitzar les dades proveïdes per les xarxes socials clàssiques com Twitter o Facebook, sinó que hauria d'ampliar l'abast a altres com poden ser Tripadvisor.
- Volum d'informació: La gran quantitat d'informació generada a la xarxa és impossible de gestionar de forma manual. La marca coca-cola únicament en el seu perfil de Facebook, té al voltant de cent milions de fans, aquest fet deixa clara la necessitat de gestionar d'una forma òptima aquest volum d'informació.

Per tal de realitzar l'anàlisi d'aquesta informació i proporcionar ajuda a les organitzacions, han nascut els sistemes de gestió de anàlisi de sentiments. Actualment existeixen multitud d'aplicacions per tal d'omplir les necessitats del mercat en aquest sector, Simply Mesured, Sysomos o la mateixa aplicació de Facebook anomenada Facebook Insights poden ajudar als Community Managers, és a dir, al personal de les organitzacions encarregats a realitzar les

comunicacions de l'empresa a les xarxes socials, a comprovar quina resposta tenen les seves interaccions amb els clients de les marques.

3.2.4.3 Sentiment analysis en l'àmbit acadèmic

Motivat per la importància no només a l'àmbit empresarial, sinó també a l'àmbit acadèmic, s'han realitzat una gran quantitat d'estudis relacionats en l'àmbit de l'anàlisi de sentiments, sent fins i tot alguns d'aquest els creadors dels terme de Sentiment analysis (Turney et al., 2002; Nasukawa et al., 2003).

El punt en comú d'aquests estudis és l'àmbit sobre el qual treballen. L'anàlisi de sentiment en realitat no és res més que la possibilitat de classificar textos en una sèrie de categories relacionades en l'entorn emocional, com poden ser la classificació de crítiques de pel·lícules en categories de valoracions positives i negatives.

El primer estudi realitzat amb algorismes de classificació proporcionava un encert de fins un 82.9% en la classificació de opinions extretes de la pàgina iMDB (Pang, Lee i Vaithyanathan et al., 2002), uns resultats esperançadors amb la utilització d'algorismes com SVN i Naive Bayes, algorismes ja amb certa antiguitat.

Altres estudis en canvi, es focalitzen en l'anàlisi de les opinions dels usuaris a les xarxes socials com Facebook o Twitter, per exemple, en analitzar la viabilitat de Twitter com a predictor del vot a les eleccions alemanyes (Tumasjan, Sprenger,Sander i Welpe et al., 2010) o bé estudis que analitzen el sentiment de diversos missatges de Twitter que continguin emoticones (Go, Bhayani i Huang et al., 2009).

Tot i que la majoria d'estudis utilitzen algoritmes clàssics com els ja mencionats Naive Bayes i SVM, algorismes més recents estan demostrant grans resultats en la seva utilització. La utilització de diverses versions de xarxes neuronals artificials, un tipus d'algorisme de Machine Learning que intenta simular el funcionament de les xarxes neuronals humanes és un dels candidats habituals en aquests articles. Concretament, la universitat de Stanford ha posat a disposició dels usuaris de la xarxa la última versió del seu propi algorisme basat en xarxes neuronals, el RTNT, per tal de poder comprovar el funcionament d'aquest en l'anàlisi de sentiment de les instàncies que se li proporcionin.

3.2.5 Algoritmes de classificació

Per tal de poder dur a terme aquest projecte, s'han hagut d'escollir una sèrie d'algoritmes per tal de realitzar l'experimentació. Aquests algoritmes no han sigut escollits a l'atzar, sinó que hi ha hagut dos factors claus que han causat la seva elecció:

- **Popularitat**: El primer i principal factor, ha sigut la utilització recurrent dels algoritmes en els articles científics més populars sobre la temàtica de classificació de textos i sentiment analysis. Per tal de trobar els articles més populars, s'ha fet servir el nombre de citacions que tenen aquests en cercadors com Google Scholar.
- **Disponibilitat**: WEKA, l'eina utilitzada per homogeneïtzar l'experimentació realitzada en aquest projecte, tot i tenir a la seva disposició una gran quantitat d'implementacions d'algoritmes, no hi són tots. Algunes implementacions com per exemple la RNTN (Socher, Perelygin, Wu, Chuang, Manning, Y. Ng i Potts et al., 2013)

no apareixen dintre del conjunt d'algorismes disponibles al ser implementacions molt recents creades pel personal de Stanford per tal de realitzar els seus estudis sobre Sentiment Analysis.

Amb aquests dos factors a tenir en compte, s'ha comprovat que els algoritmes de Naive Bayes, KNN, SVN i Random Forest han sigut utilitzats de forma recurrent en diversos articles d'importància (Joachims et al,. 1997; McCallum i Niggam et al,. 1998; Yang i Liu et al,. 1999; Liaw i Wiener et al,. 2002) . Uns altres tipus d'algoritmes utilitzats de forma habitual amb molt bons resultats han sigut diverses implementacions de xarxes neuronals. Malauradament, com ja s'ha comentat anteriorment, aquests articles utilitzen implementacions pròpies de xarxes neuronals que no existeixen a WEKA i a més, al ser la seva utilització no trivial, l'estudi requerit per implementar-ho en l'àmbit del projecte hauria requerit d'una sèrie d'hores de les quals no es disposaven.

En els següents apartats es mostrarà el funcionament intern de cadascun dels algoritmes escollits.

3.2.5.1 Naive Bayes

El primer algoritme que s'ha estudiat per la realització d'aquest projecte, és l'algoritme Naive Bayes, basat en el teorema de Bayes que formalment, mostra la probabilitat condicional d'un esdeveniment aleatori A donat B en termes de distribució de probabilitat condicional de l'esdeveniment B donat A i la distribució de probabilitat marginal de l'esdeveniment A. És a dir, permet saber la possibilitat d'un esdeveniment, basant-se en les condicions que puguin estar relacionades en aquest esdeveniment.

Matemàticament, el teorema de Bayes es pot resumir en la formula 7:

$$P(A|B\ ) = \frac{P(B|A)P(A)}{P(B)}$$

Formula 7. Teorema de Bayes

- P(A) i P(B) són les probabilitats de A i B de forma independent
- P(A|B) és la probabilitat condicional de A suposant que B sigui cert.
- P(B|A) és la probabilitat de B suposant que A és cert.

Per tal d'aclarir el funcionament d'aquest algoritme i aprofitant la seva extrema simplicitat, es mostrarà al lector un exemple bàsic sobre el funcionament d'un classificador Naive Bayes. El funcionament mostrat estarà basat en el Sentiment Analysis, on s'entrenarà el classificador amb una sèrie de crítiques fictícies de pel·lícules classificades en crítiques positives o negatives. Un cop s'hagi realitzat l'entrenament, es mostrarà com es pot classificar una nova instància utilitzant el model creat per l'algoritme.

Es disposen de les següents instàncies per entrenar el classificador:

Instància	Text	Classe
1	I loved the movie	+
2	I hated the movie	-
3	A gread movie, good movie	+
4	Poor acting	-
5	Great acting, a good movie	+

Taula 5. Dataset d'entrenament d'exemple

Utilitzant el model de bossa de paraules explicat en apartats anteriors, es poden extreure un conjunt de 10 paraules úniques del dataset d'entrenament:

Paraules úniques
<I, loved, the, movie, hated, a, great, poor, acting, good>

Taula 6. Paraules úniques extretes del dataset d'entrenament

Un cop obtingudes les paraules, es pot passar a transformar les instàncies inicials a un conjunt de vectors de paraules. El resultat d'aquesta transformació es mostra a la taula 7.

Instància	I	Loved	The	Movie	Hated	A	Great	Poor	Acting	good	Classe
1	1	1	1	1	0	0	0	0	0	0	**+**
2	1	0	1	1	1	0	0	0	0	0	**-**
3	0	0	0	2	0	1	1	0	0	1	**+**
4	0	0	0	0	0	0	0	1	1	0	**-**
5	0	0	0	1	0	1	1	0	1	1	**+**

Taula 7. Instàncies d'exemple transformades a vectors de paraules

A continuació, es mostren el conjunt de càlculs a realitzar mitjançant els valors del vector de paraules. Aquests càlculs són els que ajudaran a construir un model per tal de classificar noves instàncies en un futur.

Els primers càlculs a realitzar són per trobar el valor de la classe positiva.

$$P(+) = \frac{3}{5} = 0.6$$

A continuació, per tal de trobar la probabilitat de w_k respecte a la classe positiva s'utilitza la formula 8.

$$P(w_k|+) = \frac{n_k + 1}{n + |Vocabulary|}$$

Formula 8. Probabilitat d'un atribut respecte a una classe

On n_k és el nombre d'aparicions de l'atribut a analitzar, i n el nombre total de paraules en la classe, incloses les repetides.

Aplicant-la, es poden obtenir totes les probabilitats dels atributs tal com es pot comprovar a continuació.

$$P(I\,|+) = \frac{1+1}{14+10} = 0.0833 \qquad P(the\,|+) = \frac{1+1}{14+10} = 0.0833$$

$$P(a \mid +) = \frac{2+1}{14+10} = 0..125 \qquad P(acting \mid +) = \frac{1+1}{14+10} = 0.0833$$

$$P(hated \mid +) = \frac{0+1}{14+10} = 0.0417 \qquad P(loved \mid +) = \frac{1+1}{14+10} = 0.0833$$

$$P(movie \mid +) = \frac{4+1}{14+10} = 0.2083 \qquad P(great \mid +) = \frac{2+1}{14+10} = 0.125$$

$$P(good \mid +) = \frac{2+1}{14+10} = 0.125 \qquad P(poor \mid +) = \frac{0+1}{14+10} = 0.0417$$

Un cop obtingudes les probabilitats de la classe positiva, es poden calcular les probabilitats per la classe negativa.

$$P(-) = \frac{2}{5} = 0.4$$

$$P(I \mid -) = \frac{1+1}{6+10} = 0.125 \qquad P(the \mid -) = \frac{1+1}{6+10} = 0.125$$

$$P(a \mid -) = \frac{0+1}{6+10} = 0.0625 \qquad P(acting \mid -) = \frac{1+1}{6+10} = 0.125$$

$$P(hated \mid -) = \frac{1+1}{6+10} = 0.125 \qquad P(loved \mid -) = \frac{0+1}{6+10} = 0.0625$$

$$P(movie \mid -) = \frac{1+1}{6+10} = 0.125 \qquad P(great \mid -) = \frac{0+1}{6+10} = 0.0625$$

$$P(good \mid -) = \frac{0+1}{6+10} = 0.0625 \qquad P(poor \mid -) = \frac{1+1}{6+10} = 0.125$$

Finalment, la formula següent ens permet assignar una nova instància a una classe V:

$$V_{NB} = argmax_{v_j \epsilon V} P(v_j) \prod_{w \, \epsilon \, words} P(w \mid v_j)$$

Gràcies a aquesta formula, podem classificar noves instàncies com la que es presenta a continuació: 'I hated the poor acting'. Els càlculs següents mostren la possibilitat de que la nova instància pertanyi a una classe positiva o negativa.

Si $V_j = +$; $P(+)P(I \mid +)P(hated \mid +)P(the \mid +)P(poor \mid +)P(acting \mid +) = 6.03\ x\ 10^{-7}$

Si $V_j = -$; $P(-)P(I \mid -)P(hated \mid -)P(the \mid -)P(poor \mid -)P(acting \mid -) = 1.22\ x\ 10^{-5}$

Ja per finalitzar, si normalitzem els valors obtenim els següents resultats:

Classe	Percentatge
+	4.7
-	**95.3**

Taula 8. Resultats de la classificació

Com es pot veure clarament, l'algoritme ha classificat la nova instància passada al model com a pertanyent a la classe negativa.

3.2.5.2 KNN

KNN és l'acrònim de K-Nearest Neighbors, un algoritme que té la base del seu funcionament en la classificació de la nova instància en base a la classe més freqüent de les K instàncies més properes ja classificades.

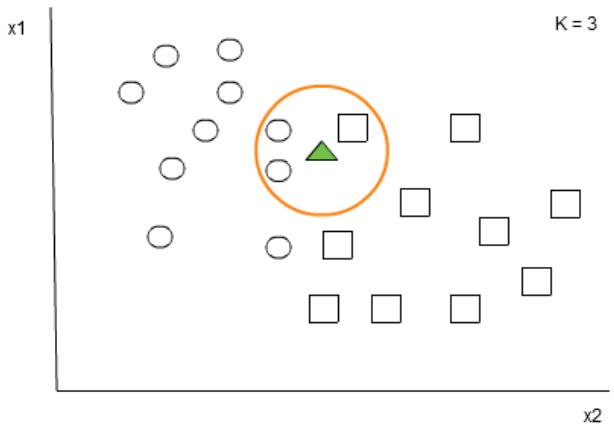
En aquest apartat, es mostrarà primerament al lector un exemple visual de la representació d'un algoritme KNN abans de proporcionar una descripció més formal. Això és degut a que la representació visual del model es realment intuïtiva i ajuda a comprendre el seu funcionament de forma realment pràcticament immediata.

3.2.5.2.1 Exemple visual

La imatge 17, mostra un clar exemple de la classificació d'una nova instància utilitzant l'algoritme KNN.

En l'exemple, es mostra una sèrie d'instàncies ja classificades, és a dir, el corpus, en dos classes diferents: cercles i quadrats. La representació d'aquestes instàncies està situada sobre un pla de dos dimensions, on cada eix representa cadascuna de les característiques de les instàncies, les quals son en aquest cas la característica genèrica x1 i x2. A continuació, s'intenta classificar una nova instància, un triangle verd utilitzant l'algoritme KNN, donant una K = 3, és a dir, s'intentarà classificar la nova instància triangle, basant-se en les classes dels tres elements més propers, el lector pot veure aquest fet en el cercle taronja de la figura, el qual engloba els tres elements més propers al triangle, sent aquests dos cercles i un quadre.

Un cop calculats quins són els tres elements més propers a la nova instància, l'algoritme únicament ha de comprovar quina és la classe amb més aparicions entre els elements més propers, així doncs, tal com es pot veure en la imatge 17, al ser el cercle la classe amb més instàncies dintre dels elements més propers, la nova instància triangle es classificarà com un cercle.



Imatge 17. Exemple d'algoritme KNN. Font: Elaboració pròpia

3.2.5.3 Definició formal

Mitjançant la imatge 10 es defineix la notació del paradigma K-NN:

		X_1	...	X_j	...	X_n	C
$(\mathbf{x}_1, c_1)$	1	x_{11}	...	x_{1j}	...	x_{1n}	c_1
	$\vdots$	$\vdots$		$\vdots$		$\vdots$	$\vdots$
$(\mathbf{x}_i, c_i)$	i	x_{i1}	...	x_{ij}	...	x_{in}	c_i
	$\vdots$	$\vdots$		$\vdots$		$\vdots$	$\vdots$
$(\mathbf{x}_N, c_N)$	N	x_{N1}	...	x_{Nj}	...	x_{Nn}	c_N
$\mathbf{x}$	$N+1$	$x_{N+1,1}$	...	$x_{N+1,j}$	...	$X_{N+1,n}$	?

Imatge 18. Taula de notació del paradigma K-NN. Font: Universidad del País Vasco

- D indica un dataset de N casos, cadascun dels quals està caracteritzat per *n* característiques, $x_1, \dots, x_n$ i una variable a predir, la classe C.
- Les N instàncies es denoten per

$$(x_1, c_1), \dots, (x_n, c_n) \; on$$

$$x_i = (x_{i,1} \dots x_{i,n}) \; per \; a \; tot \; i = 1, \dots, N$$

$$c_1 \in \{ c^1, \dots, c^n \} \; per \; a \; tot \; i = 1, \dots, N$$

- La nova instància a classificar es denota per $x = (x_1, \dots, x_n)$

Mitjançant aquesta notació ja es pot definir en pseudo-codi l'algoritme KNN:

INICI

Entrada: $D = \{(x_1, c_1), \dots, (x_N, c_N)\}$

$x = (x_1, \dots, x_n)$ nova instància a classificar

PER A TOTA instància ja classificada (x_i, c_i)

Calcular $d_i = d(x_i, x)$

Ordenar $d_i (i = 1, \dots, N)$ en ordre ascendent

Obtenir les K instàncies D_x^K ja classificades més properes a x

Assignar a x la classe més freqüent en D_x^K

FINAL

L'explicació textual de l'algoritme és la següent:

- S'introdueixen les instàncies de D en un espai euclidià multidimensional, on cada instància es representa en forma de vector en el qual cada dimensió representa una de les característiques utilitzades per classificar a aquesta, a més de la classe en la qual la instància s'ha classificat.
- A continuació es representa en aquest espai la nova instància a classificar.
- Es calculen totes les distàncies de les instàncies ja classificades amb la nova a classificar.
- S'obtenen les K distàncies més properes, la classe més freqüent entre aquestes K serà la classe assignada a la nova instància.

Amb l'algoritme i l'explicació a la vista, al lector li poden sortir diversos dubtes sobre l'algoritme, el primer és com es calcula la distància i que succeeix en cas d'empat entre classes més freqüents.

En el cas de la distància, aquesta es calcula mitjançant la formula de la distància euclidiana entre dos punts.

$$d(P,Q) = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2}$$

Formula 9. Distància euclidiana entre dos punts

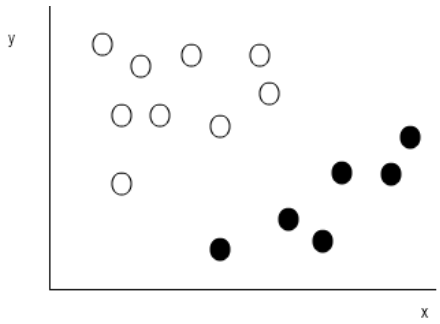
Per respondre al dubte dels empats i també en relació al càlcul de la distància, al igual que en l'algoritme Naive Bayes, no hi ha únicament una sola implementació de l'algoritme, sinó que existeixen diverses variacions. Aquestes variacions poden afectar en la forma en la que es calcula la distància i s'obté la categorització final, per exemple, hi ha variacions en les quals s'assigna un pes a la distància entre dos punts, amb la qual una instància més propera a la nova a classificar té més importància que una més llunyana. D'aquesta forma, en comptes de tenir en compte únicament el nombre predominant de classes a l'hora de realitzar la classificació, es té en compte també aquest pes.

3.2.5.4 SVM

Quan en Machine Learning es parla de SVM, s'està referint a les Support Vector Machines o màquines de vectors de suport. Aquest tipus d'algoritme, creat per Vladimir N. Vapnik al 1963, consisteix originalment en classificar una sèrie d'objectes descrits mitjançant dades numèriques en un espai n-dimensional en dues possibles classes separades per un hiperplà.

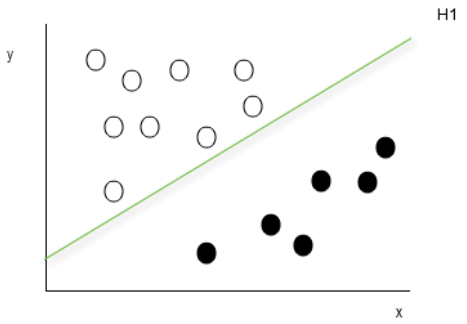
Degut a que els conceptes matemàtics per entendre el funcionament de l'algorisme són relativament complexes i que la motivació d'aquests apartats és la de mostrar el seu funcionament general, s'ha optat per mostrar el funcionament d'aquest de forma visual.

En la imatge 19 es pot veure representat un conjunt d'instàncies en un pla. Cada instància està representada per un parell d'atributs que localitzen la posició de la instància en aquest pla.



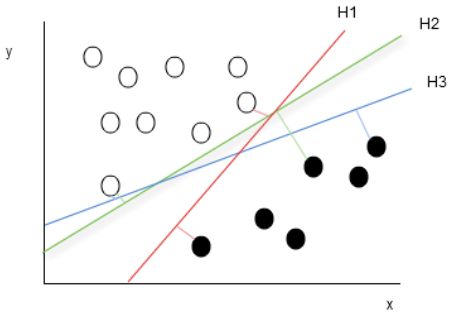
Imatge 19. Representació d'un conjunt d'instàncies de dos atributs en un pla. Font: Elaboració pròpia

L'objectiu de l'algorisme SVM és aconseguir separar les dos classes existents en aquest pla en dos parts mitjançant un hiperplà, el qual estarà representat en una línia infinita en un espai bidimensional. Un cop dividides les instàncies, qualsevol nova instància representada en aquest espai, estarà classificada en una classe o altre depenent de la banda de l'hiperplà on hagi estat situada. En la imatge 12 es pot observar aquesta situació.



Imatge 20. Representació d'un hiperplà en un espai bidimensional. Font: Elaboració pròpia

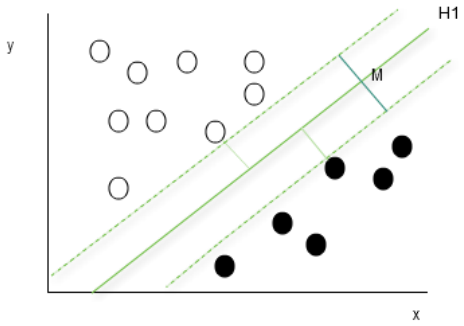
Evidentment, existeixen molts possibles hiperplans que poden separar les dos classes. La solució a quin hiperplà és el correcte l'aporta el propi SVM, el qual considera que el millor hiperplà entre tots els possibles és aquell que té un marge més ampli entre els elements més propers entre les dues classes representades. La imatge 21 mostra la representació d'alguns d'aquests hiperplans i els marges corresponents.



Imatge 21. Representació de diversos hiperplans i els seus marges. Font: Elaboració pròpia

Intuïtivament el motiu és clar, quan major distància hi hagi entre l'hiperplà i l'element més proper de les dos classes, o vectors de suport, menys possibilitats hi ha que la classificació

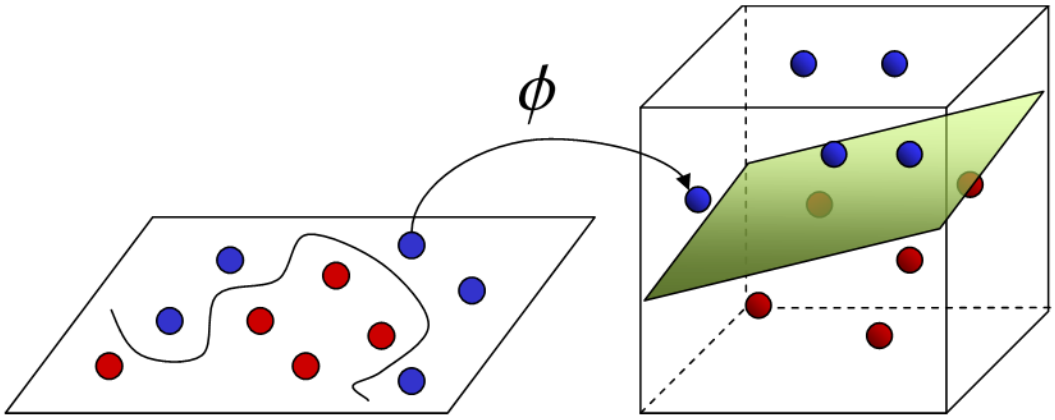
d'una nova instància sigui errònia. La imatge 22 mostra la representació d'un hiperplà òptim H1 conjuntament amb M, el qual representa el marge.



Imatge 22. Hiperplà òptim. Font: Elaboració pròpia

És evident que no tots els datasets tindran una conjunció d'entrades situades de forma tan apropiada per tal de representar un hiperplà amb tanta facilitat. Per aquest motiu, a l'any 1992 es va millorar l'algorisme amb la creació de les màquines de vectors de suport no lineals, les quals mitjançant el concepte del *kernel trick* permetien solucionar aquest problema.

Aquesta expansió de les SVM, permet la utilització de kernels, les quals són una sèrie de funcions que permeten transformar les dades disponibles en l'espai d'atributs a un espai de dimensions superior per tal de facilitar la cerca d'un hiperplà. D'aquesta forma, tot i que en l'espai dimensional original, no es pugui trobar una funció lineal per separar les dos classes, si que se'n podrà trobar una en aquest espai dimensional superior. La imatge 23 mostra la transformació nombrada.



Imatge 23. A l'esquerra, l'espai dimensional original. A la dreta, la transformació a un espai dimensional superior per trobar l'hiperplà. Font: Stackoverflow

Per finalitzar, en els problemes de classificació, tal com succeeix en aquest projecte, és possible que existeixin més de dues classes possibles a classificar les instàncies d'un dataset. En aquest cas s'utilitzen una varietat de tècniques per tal de realitzar classificacions multi-classe. La utilitzada per WEKA, és la coneguda com la tècnica del un contra un. Aquesta tècnica

consisteix en la creació de $\frac{C*C-1}{2}$ classificadors, on C és el nombre de classes existents. Cada parell de classes classificarà la instància, sent la seleccionada aquella que la majoria de classificadors hagin predit.

3.2.5.5 Random Forests

L'últim algoritme a utilitzar en aquest projecte, és també el més nou de tots els presentats. Desenvolupat inicialment per Tin Kam Ho en *Random Decision Forests* (Kam Ho et al.,1995) i ampliat per Leo Breiman i Adele Cutler en *Random Forests* (Breiman et al. 2001), aquest algoritme es podria considerar una evolució dels arbres de decisió, doncs fa ús d'aquests per tal de proporcionar un valor final. Per aquest motiu, abans de procedir a l'explicació de l'algoritme de Random Forest, es proporcionarà al lector una introducció sobre aquest tipus d'arbres.

3.2.5.5.1 Arbres de decisió

3.2.5.5.1.1 Introducció

Un arbre de decisió construeix un model de classificació en forma d'estructura d'arbre en el qual, cada node intern correspon a cadascuna de les variables del dataset, mentre que els nodes fulla corresponen a les classes sobre les quals s'està intentant classificar el corpus. L'objectiu a aconseguir és obtenir un conjunt de regles a seguir per tal de poder classificar noves instàncies.

Existeixen diversos algoritmes que implementen l'aprenentatge per arbres de decisió, entre ells hi ha l'algoritme ID3, el C4.5, el qual és una ampliació del primer, els arbres CART, entre altres.

En aquest apartat es descriurà el funcionament de l'ID3 per dos motius: el primer és la gran quantitat d'informació disponible sobre ell al ser un dels primers existents en aquest àmbit, la segona raó per la elecció és degut a la seva condició de ser l'algorisme en el qual està basat el C4.5, emprat per WEKA.

Abans de començar a explicar el funcionament de l'ID3, és important esmentar les mètriques que s'utilitzen.

3.2.5.5.1.2 Mètriques

L'aprenentatge per arbres de decisió, utilitza dos conceptes en la majoria de les seves variants: la entropia i el guany d'informació.

El concepte d'entropia representa el grau d'impuresa, el qual mesura el grau d'homogeneïtat d'un dataset. La formula per calcular l'entropia d'un dataset amb n classes es pot veure a continuació.

$$E(S) = -\sum_{i=1}^{n} p(I) * \log_2 p(I)$$

Formula 10. Entropia

En la qual p(I) representa la probabilitat de la classe I en el dataset S.

El segon concepte utilitzat en els arbres de decisió és el guany d'informació. Aquest concepte representa la capacitat que té un atribut per reduir l'entropia. Per tant, quan més alt sigui aquest valor, més informació proporcionarà l'atribut i més important serà respecte a la resta.

La formula 11 és la utilitzada per realitzar el càlcul del guany d'informació d'un atribut.

$$InfoGain(S, A) = E(S) - \sum_{v \in V(A)} \frac{|Sv|}{|S|} E(Sv)$$

Formula 11. Guany d'informació

On S és el dataset principal, Sv un subconjunt de S i E(Sv) l'entropia d'aquest subconjunt.

Un cop explicats aquests dos conceptes, es passarà a explicar el funcionament de l'algorisme ID3 amb un exemple per tal de clarificar els conceptes.

3.2.5.5.1.3 Implementació

La definició del pseudocodi de l'algoritme ID3 és la següent:

- Calcular l'entropia del dataset complert.
- Per cada atribut del corpus:
    - Dividir el dataset per cadascun dels valors que té l'atribut, és a dir, si l'atribut és color i té com a valors 'verd' i 'vermell', dividir el corpus en dos subconjunts, un en el qual l'atribut té el valor 'verd' i l'altre en el qual l'atribut té el valor 'vermell'.
    - Calcular l'entropia del nou subconjunt.
    - Calcular el guany d'informació de l'atribut.
- Comparar el guany d'informació de tots els atributs.
- Per l'atribut amb guany d'informació més elevat, utilitzar-lo com a arrel de l'arbre de decisió i cada un dels valors com una possible branca de l'arbre.
- Dividir el dataset en subconjunts per cada valor que tingui aquest atribut.
- Per cada subconjunt:
    - Si la entropia es 0, col·locar un node fulla amb el valor de la predicció.
    - En cas contrari, tornar a executar l'algoritme amb el nou subconjunt.

Un cop realitzada la execució de l'algoritme fins que totes les branques tenen fulles, es pot extreure un conjunt de regles amb les quals es poden classificar noves instàncies.

A continuació es mostrarà un exemple pràctic per tal de veure el funcionament de l'algoritme.

3.2.5.5.1.4 Exemple pràctic

Sent S el dataset mostrat a la taula 9, en el qual es mostra quin transport ha escollit una persona a l'hora de realitzar un viatge de vacances:

Atribut				Classe
Gènere	**Cotxes disponibles**	**Cost del viatge**	**Salari**	**Transport**
Home	0	Barat	Baix	Autobús
Home	1	Barat	Mig	Autobús
Dona	1	Barat	Mig	Tren
Dona	0	Barat	Baix	Autobús
Home	1	Barat	Mig	Autobús
Home	0	Normal	Mig	Tren
Dona	1	Normal	Mig	Tren
Dona	1	Car	Alt	Cotxe
Home	2	Car	Mig	Cotxe
Dona	2	Car	Alt	Cotxe

Taula 9. Dataset d'exemple

Per aplicar l'algorisme es seguiran els següents passos:

1. Calcular l'entropia del conjunt complert:

$$P(Autobús) = \frac{4}{10} = 0.4 \qquad P(Tren) = \frac{3}{10} = 0.3 \qquad P(Cotxe) = \frac{3}{10} = 0.3$$

Entropia:

$$E(S) = -0.4 * \log(0.4) - 0.3 * \log(0.3) - 0.3 * \log(0.3) = 1.571$$

2. Per cada atribut:
   - Dividir el dataset per cadascun dels valors que té l'atribut i calcular l'entropia del nou conjunt.

Gènere	Transport
Home	Autobús
Home	Autobús
Home	Autobús
Home	Tren
Home	Cotxe

Taula 10. Dataset amb l'atribut 'Gènere' amb valor 'Home'

$$E(S') = -0.6 * \log(0.6) - 0.2 * \log(0.2) - 0.2 * \log(0.2) = 1.522$$

Gènere	Transport
Dona	Tren
Dona	Autobús
Dona	Tren
Dona	Cotxe
Dona	Cotxe

Taula 11. Dataset amb l'atribut 'Gènere' amb valor 'Dona'

$$E(S'') = -0.2 * \log(0.2) - 0.4 * \log(0.4) - 0.4 * \log(0.4) = 1.371$$

- Calcular el guany de l'atribut

$$InfoGain(S, Gènere) = 1.571 - \left(\left(\frac{5}{10}\right) * 1.522\right) + \left(\left(\frac{5}{10}\right) * 1.371\right) = 0.12$$

L'exemple és anàleg pels altres atributs, per aquest motiu mostrarem directament els resultats d'aquests.

$$InfoGain(S, Cotxes\ disponibles) = 0.534$$

$$InfoGain(S, Cost\ del\ viatge) = 1.21$$

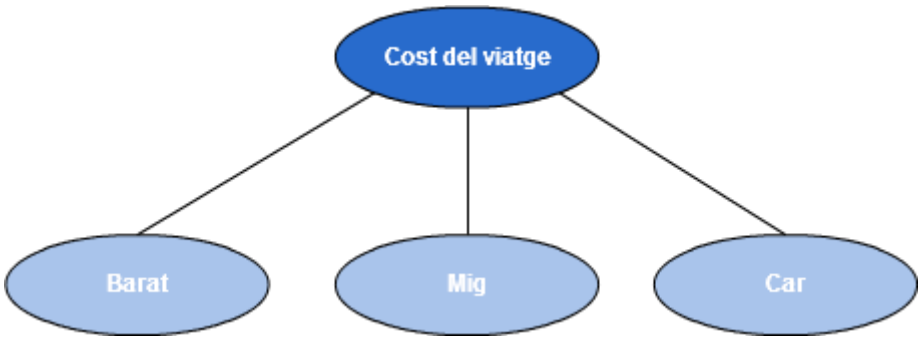
$$InfoGain(S, Salari) = 0.695$$

3. Comparar el guany de tots els atributs i escollir l'atribut amb més guany

Atributs	Guany
Gènere	0.12
Cotxes disponibles	0.534
Cost del viatge	1.21
Salari	0.695

Taula 12. Resultats dels càlculs del guany d'informació

4. Utilitzar l'atribut com arrel i dividir el dataset entre els valors d'aquest atribut.



Imatge 24. Arbre de decisió amb l'atribut cost del viatge com arrel

Atribut				Classe
Gènere	**Cotxes disponibles**	**Cost del viatge**	**Salari**	**Transport**
Home	0	Barat	Baix	Autobús
Home	1	Barat	Mig	Autobús
Dona	1	Barat	Mig	Tren
Dona	0	Barat	Baix	Autobús
Home	1	Barat	Mig	Autobús

Taula 13. Dataset amb l'atribut 'Cost del viatge' amb valor 'Barat'

Atribut				Classe
Gènere	**Cotxes disponibles**	**Cost del viatge**	**Salari**	**Transport**
Home	0	Normal	Mig	Tren
Dona	1	Normal	Mig	Tren

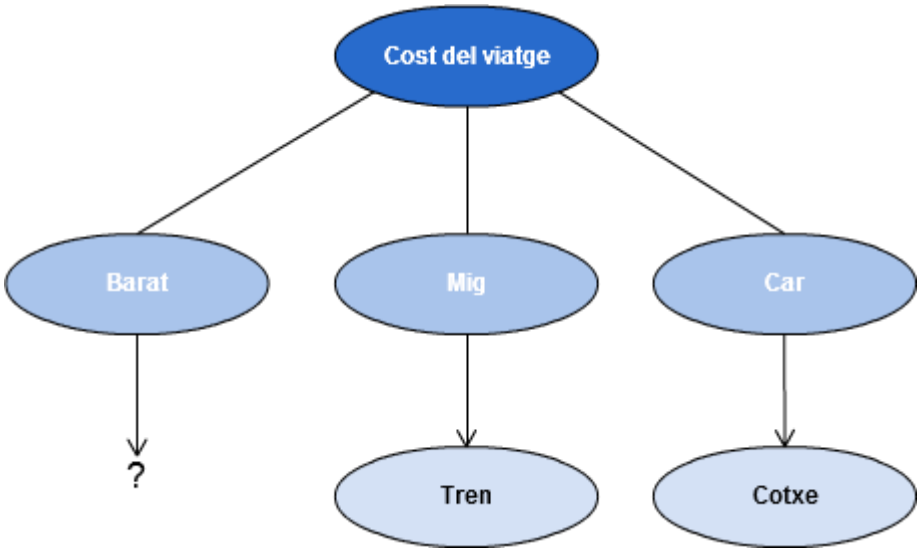
Taula 14. Conjunt amb l'atribut 'Cost del viatge' amb valor 'Normal'

Atribut				Classe
Gènere	**Cotxes disponibles**	**Cost del viatge**	**Salari**	**Transport**
Dona	1	Car	Alt	Cotxe
Home	2	Car	Mig	Cotxe
Dona	2	Car	Alt	Cotxe

Taula 15. Conjunt amb l'atribut 'Cost del viatge' amb valor 'Car'

5. Per cada subconjunt, col·locar un node fulla en cas de que l'entropia sigui 0, en cas contrari, repetir l'execució de l'algoritme.

   A simple vista es pot comprovar que els dos últims datasets tenen una entropia de 0 al ser completament homogenis, mentre que el dataset pertanyent al valor 'Barat' no compleix la condició. Per tant, els dos últims datasets se'ls hi assignarà un node fulla, mentre que en primer dataset serà utilitzat com entrada per la nova execució de l'algoritme.



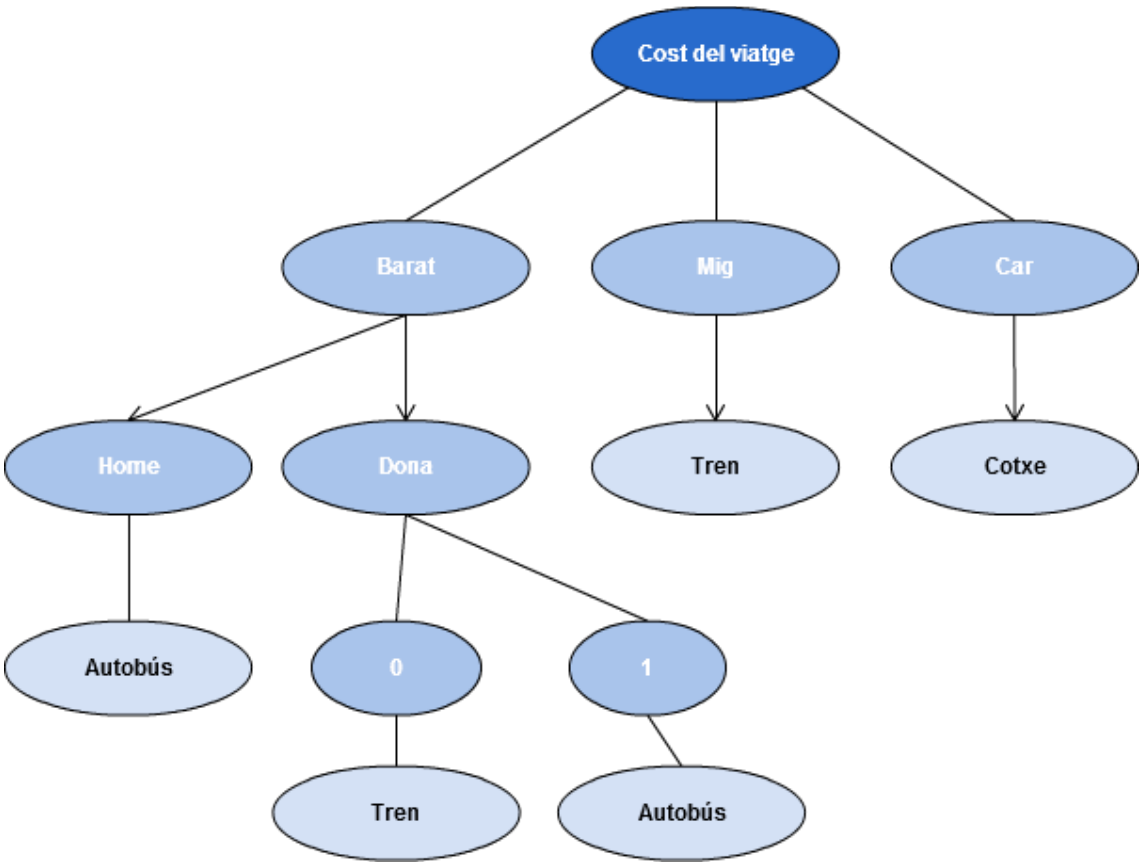
Imatge 25. Arbre de decisió després de realitzar la primera execució

La imatge 26 mostra l'arbre final després de repetir l'algorisme fins l'última execució.

Un cop amb l'arbre finalitzat, seguint la seva execució des de l'arrel es poden predir nous valors. La taula 16 en mostra un exemple pràctic.

Atribut				Classe
Gènere	**Cotxes disponibles**	**Cost del viatge**	**Salari**	**Transport**
Home	1	Normal	Alt	Tren
Home	0	Barat	Mig	Tren

Taula 16. Predicció de noves instàncies en un arbre de decisió



Imatge 26. Arbre de decisió final

3.2.5.5.1.5 Millores en l'algorisme C4.5

Degut a que l'algorisme C4.5, és la versió utilitzada en l'aplicació WEKA, en aquest apartat s'esmentaran algunes de les millores implementades.

- Utilització d'atributs continus i discrets.
- Utilització de dades d'entrenament amb valors d'atributs perduts.
- Gestió d'atributs amb diferents costos.
- Poda de l'arbre després de la creació.

3.2.5.5.2 Funcionament de Random Forest

Un cop mostrat el funcionament de l'aprenentatge per arbres de decisió, la explicació sobre el funcionament de Random Forest resulta trivial.

Random Forest per tal de realitzar les seves prediccions, divideix el dataset d'entrada en diversos conjunts de forma aleatòria, aplicant per cadascun d'aquests subconjunts l'algorisme d'un arbre de decisió. Un cop tots els arbres retornen el resultat, s'aplica la regla de la majoria

per obtenir la classificació desitjada. En cas d'empat, es seleccionarà aleatòriament el resultat entre els candidats amb major nombre de vots.

3.2.6 WEKA

Tot i que en un primer moment, una de les tasques del present projecte consistia en la recerca d'algoritmes per Machine Learning i la seva implementació de forma individual a posteriori per tal de realitzar la experimentació, no va ser fins més endavant quan es va descobrir WEKA, una de les eines utilitzades per la comunitat científica per tal d'homogeneïtzar tot tipus d'experimentacions en el camp del Machine Learning.

Per aquest motiu, en els punts posteriors, es presentarà al lector una introducció sobre WEKA en la qual es mostrarà què és, quina utilitat té, a més de diversos conceptes bàsics els quals seran necessaris per tal d'entendre l'experimentació que es mostrarà en punts posteriors del document. S'ha de tindre en compte però, que la descripció mostrada estarà basada en les parts de l'aplicació utilitzades durant el transcurs del projecte, degut a la complexitat d'aquesta, tractar de descriure totes les possibilitats que ofereix requeriria d'un temps molt elevat del qual no es disposa per l'abast del propi projecte.

3.2.6.1 Introducció a WEKA

WEKA és una eina creada per la universitat neozelandesa de Waikato. Aquesta eina, en realitat no és res més que un seguit d'algoritmes de Machine Learning conjuntament amb un seguit d'eines per tal d'ajudar a la experimentació de les dades passades als algoritmes. Aquestes eines poden ser eines per visualitzar dades, preprocessament de les dades d'entrada, diversos tipus de validació de dades un cop realitzada la experimentació, entre molts altres. Tot això unit amb una interfície gràfica per tal d'ajudar a l'usuari a realitzar qualsevol tasca.

Pels usuaris més experts o amb diferents necessitats, WEKA també ofereix per descarregar una llibreria Java, amb la qual qualsevol usuari amb nocions de programació pot aprofitar les avantatges que ofereix el sistema en el seu propi codi.

En el següent apartat, es mostrarà al lector el mètode utilitzat per WEKA per tal de llegir les dades requerides per l'experimentació.

3.2.6.2 Arxius ARFF

Per tal de poder utilitzar al conjunt d'eines i algoritmes que ofereix, no se li pot passar qualsevol tipus de dades sense preparar, sinó que s'ha de realitzar una transformació d'aquestes al format que accepta WEKA per realitzar la lectura d'aquestes.

Els arxius resultants d'aquesta transformació són els arxius ARFF, que en realitat no són res més que arxius de text amb un format concret tal com podrien ser els arxius CSV.

Per tal de transformar l'arxiu amb el conjunt d'instàncies disponibles a un arxiu ARFF, s'ha de crear un arxiu de text el qual es separarà en dos seccions. Una secció de capçalera i una de dades.

3.2.6.2.1 Secció de capçalera d'un arxiu ARFF

La secció de capçalera d'un arxiu ARFF conté el nom de la relació i un llistat d'atributs, per mostrar la seva composició concreta es presenta la imatge 27.

```
% 1. Title: Iris Plants Database
%
% 2. Sources:
%      (a) Creator: R.A. Fisher
%      (b) Donor: Michael Marshall (MARSHALL%PLU@io.arc.nasa.gov)
%      (c) Date: July, 1988
%
@RELATION iris

@ATTRIBUTE sepallength  NUMERIC
@ATTRIBUTE sepalwidth   NUMERIC
@ATTRIBUTE petallength  NUMERIC
@ATTRIBUTE petalwidth   NUMERIC
@ATTRIBUTE class        {Iris-setosa,Iris-versicolor,Iris-virginica}
```

Imatge 27. Capçalera d'un ARFF. Font: WEKA

En els següents punts es mostrarà el contingut de cada camp:

- **Comentaris**: Totes les línies amb un caràcter '%' al inici són camps de comentaris descriptius que no es tindran en compte a l'hora de llegir l'arxiu per part de WEKA.
- **RELATION**: És la nomenclatura que utilitza WEKA per indicar el títol del dataset.
- **ATTRIBUTE**: En aquesta sèrie de camps s'indicaran els tipus de valors que conté el dataset. Cada atribut pot ser de diversos tipus tal com es mostra a la taula 17. A l'exemple mostrat, es disposa de cinc atributs, quatre numèrics i un enumerat, que serviran per identificar cada instància del corpus.

Tipus d'atribut	Descripció
Numeric	Els atributs d'aquest tipus poden ser reals o enters.
Enumerat	Tipus d'atribut que pot correspondre a una sèrie de valors indicats entre els caràcters '{}'
String	Atributs que són una cadena de caràcters. Els missatges dels Community Managers utilitzats a l'experimentació són d'aquest tipus.
Date	Tipus d'atribut utilitzat per indicar dates.

Relational	Tipus d'atribut que conté altres atributs. El següent exemple mostra una possible definició:
	`@attribute bag relational`
	`  @attribute f1 numeric`
	`  ...`
	`  @attribute f166 numeric`
	`@end bag`

Taula 17. Tipus d'atributs en un arxiu ARFF

3.2.6.2.2 Secció de dades en un arxiu ARFF

La secció de dades d'un arxiu ARFF té una definició molt més senzilla que la capçalera. A continuació es mostra un exemple de la secció de dades.

```
@DATA
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
```

Imatge 28. Secció de dades d'un arxiu ARFF. Font: WEKA

Tal com el lector podrà observar, la secció de dades comença amb la declaració de la cadena de text @DATA, tot el contingut posterior a aquesta etiqueta, WEKA l'entendrà com les instàncies que pertanyen al dataset.

En l'exemple de la imatge, es poden visualitzar tres instàncies, amb els atributs declarats a la capçalera separats per comes.

3.2.6.3 Tractament de dades d'entrada

Les dades que s'utilitzaran com entrada dels algoritmes, molt sovint no seran aquelles que s'han obtingut directament de la font original, sinó que han de passar per una sèrie de processos per tal de transformar els datasets inicials a uns diferents per tal de poder utilitzar els algoritmes de classificació al seu màxim rendiment. En aquest apartat es mostraran algunes de les múltiples opcions disponibles a WEKA per tal de realitzar aquest processos, donant èmfasi a aquelles utilitzades en el transcurs de l'experimentació.

3.2.6.3.1 String to Word Vector

El primer filtre a explicar és el més important, doncs sense ell no es pot realitzar cap tipus d'experimentació degut a que els algoritmes utilitzats en la realització de l'experimentació no accepten dades d'entrada en format string. El motiu d'aquest fet és degut a que aquest no proporciona prou informació, doncs WEKA n'analitza el seu conjunt i no pas el valor més reduït d'aquest text, la paraula, amb l'adició de ser incapaç de poder trobar alguna relació entre elles.

Per aquest motiu WEKA incorpora el processament el filtre anomenat *StringToWordVector.* Aquest filtre permet convertir les cadenes de text existents en els datasets a atributs de tipus numèric. Aquest procés es pot realitzar de diverses formes, WEKA per defecte converteix cada paraula d'una instància del dataset en un atribut nou del dataset, mentre que a les dades, en comptes d'aparèixer la paraula com a tal, simplement es denota la seva aparició o no en la instància corresponent. A continuació es mostra un exemple d'un dataset bàsic per tal de que es pugui veure la transformació.

```
@relation testData

@attribute text String
@attribute category {I,E}


@data
"Adeu Paraula Paraula",E
"Hola Paraula",E
```

Imatge 29. Dataset d'exemple

En la imatge 29 es pot observar un simple dataset del qual existeixen dos instàncies amb tres i dos paraules respectivament, classificades en una mateixa categoria 'E'. Un cop s'utilitza el filtre *StringToWordVector* amb els seus valors per defecte, el dataset resultant es pot veure a continuació:

```
@relation TestStringToWordVector

@attribute category {I,E}
@attribute Adeu numeric
@attribute Hola numeric
@attribute Paraula numeric

@data
{1 1,3 1}
{2 1,3 1}
```

Imatge 30. Dataset després d'utilitzar StringToWordVector

Tal com es pot observar, en el nou dataset cada paraula s'ha convertit en un atribut nou, mentre que a la part de dades, per cada instància s'indica la existència o no de la paraula en la instància corresponent. En el cas concret de la primera instància, el primer i tercer atributs apareixen en la instància, sent aquests atributs les paraules 'Adeu' i 'Paraula'.

Evidentment, el filtre permet moltes més possibilitats en la seva configuració. Tot i que aquest projecte no té intenció de ser un manual de l'aplicació WEKA, per la seva importància es detallaran al lector algunes de les opcions més importants:

- **Freqüència de la paraula**: Es pot mostrar la freqüència d'un atribut en una instància.
- **Transformació TF-IDF**: Mitjançant la opció IDFT, es pot configurar el filtre per tal de que realitzi la transformació TF-IDF, és a dir, la raresa del terme en qüestió amb relació al conjunt de totes les instàncies del dataset.
- **Guardar els N primers atributs per classe**: Per tal de limitar el nombre d'atributs a guardar es pot demanar al filtre que guardi els N primers atributs que troba en cada classe. Aquest filtre pot ser útil en cas de tenir una quantitat molt elevada d'atributs.
- **Stemmer**: El filtre també permet passar un *stemmer* les paraules, és a dir, permet agafar l'arrel de la paraula per tal de no haver de diferenciar singular o plural, o les diferents formes verbals. Un exemple de processament per stemmer seria transformar la paraula 'cantava' en 'cantar'.
- **Stop Words**: WEKA permet seleccionar una sèrie de paraules prohibides per ser seleccionades com atribut. Paraules molt comuns com 'the', 'and', etc... poden aportar

poc valor a la instància a classificar i podria arribar a ser desitjable eliminar-les per tal de millorar la eficiència del processament.

- **Tokenizer**: Els *tokenitzadors* permeten a WEKA separar la instància inicial en atributs, el valor per defecte és convertir cada paraula en un atribut, tot i així, aquesta opció permet la utilització de n-grams, és a dir, utilitzar conjunts de N paraules com atribut en comptes d'utilitzar-ne una sola. Tot i que la utilització de n-grams ha demostrat ser efectiva en la classificació de textos (Cavnar et al, 1994), hi ha experimentacions que difereixen dels resultats obtinguts (Hidalgo et al, 2013).

3.2.6.3.2 Attribute Selection

Tant per la enorme quantitat d'atributs que es poden arribar a obtenir en certs datasets, com per la complexitat d'alguns algoritmes de Machine Learning, tècniques com la selecció d'atributs explicades en apartats anteriors poden arribar a ser necessàries per tal de millorar la eficiència de la experimentació a realitzar. Per aquest motiu, WEKA incorpora una sèrie d'eines en el seu codi per tal de que qualsevol usuari pugui aplicar aquestes tècniques de forma automàtica als seus datasets.

A l'aplicació, segons la fons consultada es poden trobar fins a dos tipus de classificació de mètodes de selecció d'atributs (Witten, Hall i Frank et al., 2011). El primer tipus de classificació consisteix en organitzar els mètodes oferts per mètodes d'avaluació d'un conjunt d'atributs i mètodes d'avaluació d'atributs individuals, mentre que el segon és la classificació en la qual els mètodes s'organitzen entre mètodes de filtratge i mètodes d'embolcallament.

Addicionalment, per tal de realitzar aquesta selecció, WEKA requereix dos paràmetres durant l'execució de la selecció d'atributs. El primer, és la selecció del mètode d'avaluació d'atributs, mentre que el segon paràmetre és el mètode de cerca.

En els següents subapartats es descriuran de forma breu les possibilitats que ofereix l'aplicació sobre els mètodes de selecció d'atributs individuals o de conjunts d'atributs, centrant l'explicació en els utilitzats durant la realització del projecte i en el significat dels conceptes de paràmetres utilitzats en els paràgrafs anteriors.

3.2.6.3.2.1 Mètodes d'avaluació individual

Els mètodes d'avaluació individual a WEKA realitzen una ordenació dels atributs mitjançant la informació que aporten aquests en una instància del dataset i eliminant aquells que aporten poca o nul·la informació a la categorització final d'aquesta.

L'únic mètode de cerca possible a utilitzar a WEKA quan es vol utilitzar aquest tipus d'avaluació, és el mètode de cerca *Ranker*, el qual crea una classificació dels atributs utilitzats durant la selecció i descarta a tots aquells que no superen el llindar indicat. El representant d'aquest tipus d'avaluació i l'utilitzat en el projecte per oferir bons resultats en la classificació de textos és l'avaluador *InfoGainAttributeEval* (Yang i Pedersen et al,. 1997), el qual avalua cada atribut respecte al guany d'informació que proporciona respecte a la classe.

Existeixen altres mètodes d'avaluació individual d'atributs com el *ReliefAttributeEval* o bé *GainRatioAttributeEval*, entre molts altres, els quals es diferencien principalment per la forma amb la qual avaluen els atributs.

3.2.6.3.2.2 Mètodes d'avaluació de subconjunts d'atributs

Els mètodes d'avaluació de subconjunts d'atributs, permeten seleccionar un conjunt d'atributs del dataset, realitzar la avaluació d'aquest conjunt i guardar els atributs amb millor resultat. Al igual que en l'avaluació individual d'atributs, és necessari seleccionar entre el mètode d'avaluació i el mètode de cerca.

En el cas de mètodes d'avaluació de subconjunts, existeixen avaluadors com el *CfsSubsetEval*, que avalua un subconjunt d'atributs considerant l'habilitat predictiva individual de cada variable, així com el grau de redundància entre elles o bé altres com el *WrapperSubSetEval*, el qual fa servir algoritmes de classificació de Machine Learning per avaluar un conjunt d'atributs.

Els mètodes d'avaluació de subconjunts d'atributs també requereixen de la introducció d'un mètode de cerca per tal de realitzar el procés de selecció d'atributs. WEKA disposa de dos opcions per poder realitzar aquesta cerca:

- *GreedyStepwise*: Mètode que realitza una cerca mitjançant un algoritme voraç per trobar quins són els millors conjunt d'atributs del dataset.
- *BestFirst*: El segon mètode de cerca que utilitza WEKA és un mètode molt semblant a l'anterior al realitzar de forma similar, una cerca mitjançant un algoritme voraç, concretament un *Hill-Climbing*. La principal diferència entre els dos mètodes, és la capacitat de *backtracking* del mètode BestFirst, és a dir, la capacitat de comprovar altres possibilitats en l'espai de cerca descartades a priori per tal de cercar una solució millor, en comptes de quedar-se en la primera solució trobada.

Per finalitzar, cal esmentar que tots els avaluadors nombrats anteriorment amb l'excepció de l'avaluador *WrapperSubSetEval*, formen part dels coneguts com mètodes de filtratge. Aquest últim forma part dels mètodes d'embolcallament, els quals fan servir algoritmes classificadors per realitzar la selecció d'atributs.

3.2.6.3.3 Altres tècniques de preprocessament

En els apartats anterior s'han mostrat dos de les principals tècniques a utilitzar en el preprocessament dels datasets abans de realitzar qualsevol tipus d'experimentació, però tot i que hagin sigut els utilitzats durant l'experimentació, no són els únics dels quals disposa WEKA. L'aplicació disposa de mètodes per canviar tipus d'atributs, com la possibilitat de discretitzar atributs numèrics mitjançant *Discretize* o normalitzar-los mitjançant *Normalize*, o bé la possibilitat d'afectar l'ordre dels paràmetres del mateix dataset utilitzant *ReOrder*, entre molts altres.

Per finalitzar, tots els filtres anteriors es poden aplicar en sèrie gràcies a la possibilitat que ofereix *MultiFilter*. Aquest filtre pot resultar especialment útil en cas de que sigui necessari aplicar una gran quantitat de filtres al dataset.

3.2.6.4 Experimentació

Tot el procés anterior és produeix per la necessitat de proporcionar als algoritmes la informació més útil possible per tal de que aquests puguin realitzar l'aprenentatge de la forma més eficient possible.

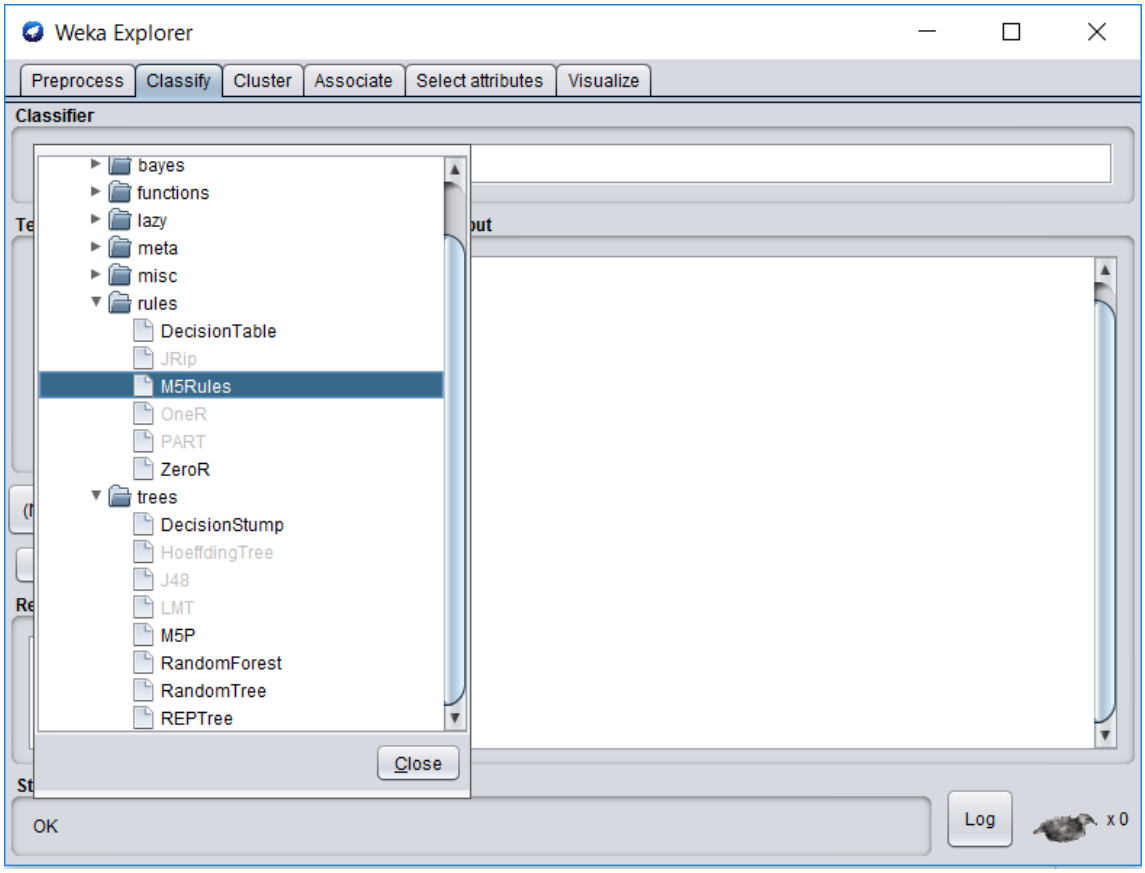
Durant aquesta etapa, WEKA ofereix tota una sèrie d'eines per tal de que l'usuari pugui realitzar l'experimentació, com la posterior etapa de validació de resultats.

En els següents apartats es mostrarà al lector quines són les eines ofertes.

3.2.6.4.1 Algoritmes classificatoris

Com ja s'ha mencionat anteriorment, una de les principals avantatges de l'aplicació, és la possibilitat d'executar un gran nombre d'algoritmes sense que l'usuari que realitza l'experimentació s'hagi d'amoïnar de la implementació d'aquests. Gràcies a aquesta característica, amb un únic dataset, l'usuari pot utilitzar-lo d'entrada a les desenes d'algoritmes dels quals disposa l'aplicació.

No tots els algoritmes tenen el mateix tipus d'entrada, és a dir, no es podrà utilitzar el mateix corpus per a tots els algoritmes disponibles a l'aplicació. Alguns algoritmes acceptaran textos com a atributs d'entrada, altres només acceptaran atributs numèrics o bé una mescla entre els dos. Per tal de facilitar aquest coneixement, WEKA indica clarament de forma visual quins són els algoritmes disponibles un cop s'ha seleccionat un arxiu arff com entrada d'aquest.



Imatge 31. Algorismes de classificació disponibles

Per finalitzar aquest apartat, cada algorisme del qual es disposa té una sèrie de opcions de personalització per tal de que l'usuari pugui millorar l'eficiència d'aquest. La configuració de la K o el mètode de mesurament de la distància entre veïns són exemples clars de les opcions de configuració disponibles en cas de seleccionar un algoritme de tipus KNN.

3.2.6.4.2 Validació de l'experimentació

Un cop seleccionat l'algorisme, WEKA ofereix diverses opcions per tal de avaluar el comportament d'aquest. Les eines en qüestió són:

- Avaluació mitjançant el propi dataset d'entrenament.
- Avaluació mitjançant un dataset de proves.
- Validació creuada.
- Separació per percentatge.

Els següents subapartats descriuran breument en que consisteixen cadascuna d'aquestes tècniques.

3.2.6.4.2.1 Avaluació mitjançant el dataset d'entrenament

El primer dels casos és també el menys indicat per proporcionar una predicció sobre el rendiment de l'algoritme. Aquesta tècnica consisteix en avaluar el model creat per l'algoritme, utilitzant com a dataset de proves, el mateix dataset utilitzat per l'entrenament. La problemàtica d'aquesta tècnica és evident, doncs al utilitzar les mateixes instàncies tant en l'entrenament com en les proves, l'algoritme no té cap dificultat en predir els resultats.

3.2.6.4.2.2 Avaluació mitjançant un dataset de proves

Aquest tipus d'avaluació produeix uns resultats més reals que l'anterior. Per utilitzar-la es requereixen dos corpus, el primer, el dataset d'entrenament s'utilitzarà per tal d'alimentar l'algoritme de classificació a utilitzar, mentre que el segon, el dataset de proves, s'utilitzarà per comprovar el rendiment que té l'algoritme al passar-li un conjunt d'instàncies noves.

Les avantatges d'aquesta tècnica són evidents, els resultats obtinguts de l'avaluació seran molt més propers al rendiment real que pot tenir l'algorisme al proporcionar-li un conjunt de dades completament diferents a les utilitzades per entrenar l'algoritme.

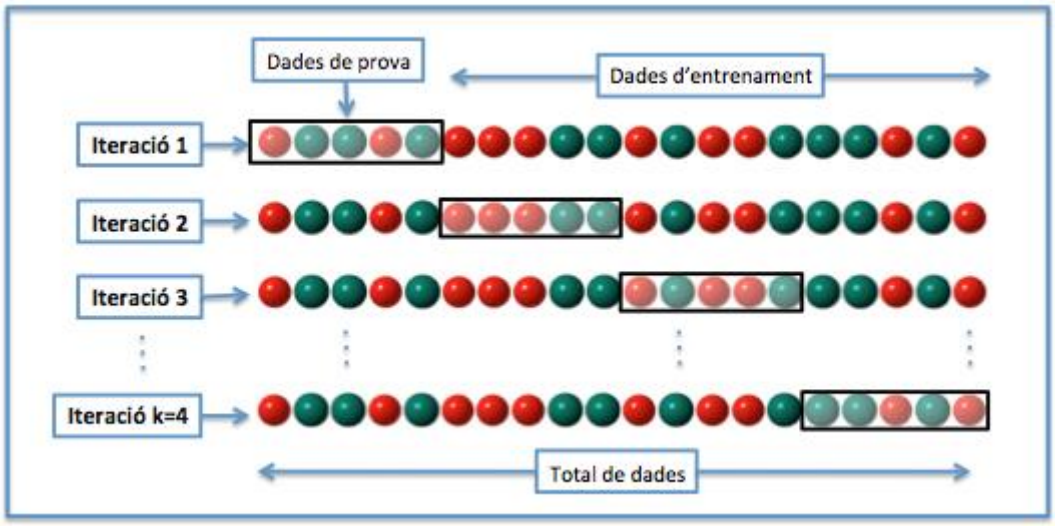
Per altra banda, la problemàtica de la representativitat de les dades dels datasets es duplica, doncs al existir dos conjunts de dades diferents es duplicarà la possibilitat de que el conjunt de dades dels corpus no sigui representatiu del problema a solucionar, amb el qual els resultats obtinguts poden no ser els esperats.

Finalment, per tal d'utilitzar aquest tipus de tècnica a WEKA, és necessari que tant el dataset d'entrenament com el de proves tinguin els mateixos atributs. Aquesta limitació es pot solucionar fàcilment utilitzant el procés de *Batch filtering*, un paràmetre utilitzat en alguns filtres com *StringToWordVector* amb el qual realitzaran el procés de transformació en ambdós datasets al mateix temps, compartint els atributs entre els dos datasets per tal d'homogeneïtzar-los.

3.2.6.4.2.3 Validació creuada

La tècnica de la validació creuada, o *cross-validation* és una de les més utilitzades conjuntament amb la tècnica anterior i considerada un dels estàndards a l'hora de comprovar la precisió d'un model al poder-se fer servir sense tenir una gran quantitat de dades.

Aquesta tècnica consisteix en dividir de forma aleatòria el dataset d'entrenament en una sèrie de *folds* o plecs, sent 10 el nombre habitual. A continuació, s'utilitzen nou dels deu plecs creats per entrenar un model, fent servir el restant com a dataset de proves. Aquest procés es repeteix tantes vegades com plecs s'hagin creat.



Imatge 32. Validació creuada amb 4 plecs. Font: Wikipedia

Un cop s'ha completat el procés, es realitza una mitja dels resultats obtinguts per cada iteració. Tot i que aquest sigui el procés estàndard, WEKA té la particularitat de executar una vegada addicional el procés per crear el model final, utilitzant en aquesta nova iteració el conjunt complet del dataset d'entrenament.

En aquest cas al igual que la tècnica anterior, l'algorisme no és entrenat amb les dades de proves, amb el qual els resultats presentats tendeixen a ser més acurats que els presentats en la tècnica d'avaluació mitjançant el corpus d'entrenament.

Per finalitzar aquest apartat, és important esmentar el concepte d'estratificació. Al utilitzar una divisió aleatòria per tal de crear cadascun dels plecs mitjançant els quals s'executarà la validació creuada, és altament probable que la distribució de les classes dintre de cada plec sigui completament irregular. Per aquest motiu es va crear la validació creuada estratificada, tècnica utilitzada per defecte per WEKA que té la avantatja de realitzar una distribució el més uniforme possible de les classes en cada un dels plecs existents.

3.2.6.4.2.4 Separació per percentatge

L'última tècnica proporcionada per l'aplicació per tal d'avaluar un algoritme és el *Percentage-split* o separació per percentatge.

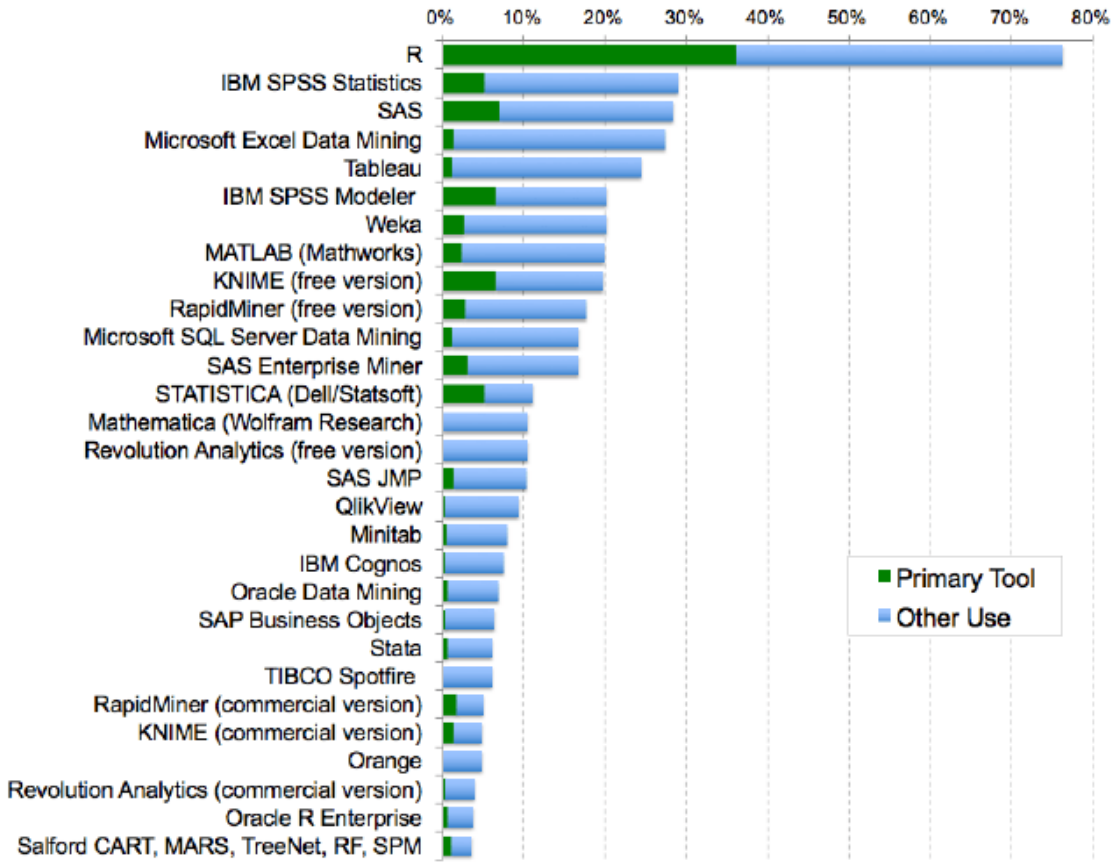
La tècnica del *percentatge-split* consisteix en la divisió aleatòria de les instàncies del corpus d'entrenament a un percentatge indicat per paràmetre. Un cop dividit, el percentatge indicat del corpus indicat serà utilitzat com a dataset d'entrenament, mentre que el restant serà utilitzat per validar els resultats obtinguts de l'entrenament.

Les avantatges d'aquesta tècnica són similars a les dos anteriors al utilitzar noves dades de prova per realitzar la validació del model creat per l'algoritme i el corpus d'entrenament.

3.2.6.5 Eines similars a WEKA

WEKA no és l'única eina existent per tal de realitzar experiments basats en Machine Learning. Existeixen diverses alternatives en el mercat actual, tan de forma gratuïta com de pagament. Algunes de les eines gratuïtes més populars són RapidMiner i el llenguatge R en conjunció amb alguna GUI, mentre que per part de les eines de pagament, SAS i SPSS propietat de IBM són les més utilitzades.

Aproximadament cada any, Rexer Analytics, una empresa dedicada a la consultoria de CRM, anàlisis avançat i descobriment del coneixement, realitza una enquesta dirigida a tots els professionals de l'àmbit de la mineria de dades, en la qual entre altres qüestions, intenta donar resposta a la pregunta de quines són les eines més utilitzades pels professionals d'aquesta àrea. La imatge 33 mostra els resultats d'aquesta enquesta.



Imatge 33. Utilització de les diverses eines de mineria de dades pels professionals del sector. Font: Rexer Analytics

Amb els resultats anterior, el lector es pot preguntar les raons de la utilització de WEKA en vers alguna de les suites de R. Per aquest motiu, en els següents punts es mostraran els fets que van propiciar aquesta decisió.

- Abast del projecte: Degut a la nul·la experiència de l'autor en quant a tota l'àrea de Machine Learning, el temps requerit per realitzar un estudi de les principals eines

disponibles i utilitzar la més adequada, hauria tingut un cost temporal massa elevat degut a la gran complexitat d'aquest tipus de sistemes.

- Descobriment de les eines existents: La existència de eines per facilitar l'experimentació dels algoritmes de Machine Learning no va succeir fins un cop avançat el projecte. Així doncs, un cop es va comprovar que WEKA tenia les capacitats suficients per complir els requisits del projecte, es va decidir la seva utilització al no tenir la necessitat de buscar més alternatives.
- Capacitat de personalització: Tot i que WEKA disposi d'una GUI per tal de facilitar la feina a l'usuari, mitjançant el llenguatge Java es pot realitzar qualsevol tipus de d'aplicació de Machine Learning utilitzant la llibreria que WEKA fica a disposició dels usuaris. R, al ser un llenguatge de programació té capacitats semblants, però en aquest cas, la facilitat d'utilització de Java i la experiència dels estudiants de la UPC en aquest llenguatge, permet facilitar la realització d'un potencial projecte davant de la seva alternativa.

4 Planificació i estudi econòmic

En aquest punt es mostrarà la planificació temporal i econòmica del projecte.

En el punt de planificació temporal apareixeran una breu descripció de les tasques realitzades, una taula on s'indicarà la duració de cadascuna d'elles, les contingències sorgides en la realització del projecte i finalment els diagrames de Gantt corresponent a la planificació inicial mostrada en l'estudi previ i la planificació real amb la qual s'ha finalitzat el projecte.

Per l'estudi econòmic o planificació econòmica es mostraran els costos materials, humans i ocupacionals en el supòsit de que aquest projecte és realitzes en un entorn laboral real.

4.1 Planificació temporal

La planificació d'aquest projecte ha sigut complicada pel nombre de contingències aparegudes durant el transcurs de la seva realització. Aquestes han provocat un greu retràs en l'entrega planificada del projecte, convertint un projecte iniciat el 29 d'abril del 2014 i planificat per entregar-se al març del 2015, en un projecte amb una entrega de memòria planificada per setembre del 2016.

La diferència entre ambdós planificacions es pot observar fàcilment mitjançant tant la imatge 34, on es mostra la planificació inicial mostrada en el treball previ, com en les imatge 35 i 36, que mostren la planificació final del projecte.

Aquest retràs, tal com ja s'ha esmentat s'ha degut a la sèrie de problemàtiques comentades a continuació:

- S'ha hagut de crear una aplicació des de cero per tal de poder extreure les dades necessàries per realitzar estudis en comptes d'utilitzar-ne una ja creada.
- La necessitat de realitzar una assignatura no prevista en la planificació inicial.
- La realització d'un curs durant el més de febrer també ha contribuït a aquest retard.
- En l'àmbit professional de l'estudiant hi ha hagut dos etapes que han contribuït al retràs.
    - La primera un augment de les hores treballades durant la jornada parcial de l'estudiant a inicis del 2015. Aquest augment va provocar un desajust important en les hores disponibles per la realització del projecte en la primera entrega planificada a març de 2015.
    - La segona etapa es va iniciar a partir de juny de 2015, moment en que aquesta jornada parcial es va convertir en jornada completa. Aquest canvi ha minimitzat al màxim la quantitat d'hores disponibles per la realització del projecte degut a l'esforç requerit durant la jornada laboral, limitant pràcticament la seva realització a caps de setmana.

Un cop explicades les contingències i abans de procedir al detall de la planificació, s'exposaran a continuació de forma breu, un llistat amb els grups de tasques que s'han realitzat durant el transcurs del projecte.

- **Introducció**: Aquest primer grup correspon a la introducció dels tutors a l'estudiant a l'àmbit del projecte.

- **Investigació API**: Un cop iniciat el projecte, el primer pas ha sigut realitzar un estudi sobre el funcionament de les APIs de Facebook i Twitter per comprendre el seu funcionament.
- **Investigació Machine Learning**: En aquest punt, ha calgut investigar l'àmbit del Machine Learning per tal de poder començar a introduir-se en ell. Aquesta introducció, a més d'aprendre alguns conceptes introductoris serviria per poder aprendre quins algorismes s'utilitzaven per tal de realitzar la categorització de textos.
- **Investigació Sentiment Analysis**: La investigació d'anàlisi de sentiments, és pràcticament anàloga al punt anterior. Un cop introduït en la matèria mitjançant la lectura d'un conjunt d'estudis científics, s'han hagut de cercar algorismes per tal de realitzar una possible experimentació. En aquest punt, es va descobrir que tan els algorismes utilitzats per la categorització de textos com els utilitzats per l'anàlisi de sentiments, són els mateixos al ser aquest últim un subconjunt dels problemes de classificació de textos.
- **Investigació WEKA**: Un cop avançat el projecte i descobert l'eina WEKA que facilitaria la execució de l'experimentació plantejada, s'ha hagut de dedicar un conjunt d'hores per descobrir el seu funcionament per tal de poder realitzar les tasques necessàries.
- **Desenvolupament eines extracció i guardat de dades**: Aquest apartat correspon al conjunt d'etapes necessàries per tal de poder realitzar la implementació del programa de extracció de dades.
- **Experimentació Machine Learning**: El punt d'experimentació mostra totes les tasques requerides per cadascun dels passos realitzats per tal de poder conèixer la viabilitat dels algorismes trobats en els apartats anteriors.
- **Documentació**: Aquest conjunt de tasques correspon a les tasques requerides per tal de redactar la informació trobada durant la investigació del projecte. Els principals artefactes utilitzats per mostrar aquesta informació són el treball previ i la memòria final del PFC.
- **Control de projecte**: Com tot PFC, s'han requerit d'una sèrie de sessions de control per part dels tutors per tal de poder orientar a l'alumne en la consecució dels objectius previstos. Aquest grup de tasques contempla aquestes sessions.
- **Defensa del projecte**: L'última tasca correspon a l'entrega i defensa del projecte per part de l'alumne.

A continuació, es presenta al lector una taula amb les dades detalles de les tasques realitzades, les dates d'inici, final, les dependències entre les tasques en el diagrama de Gantt i finalment, les hores dedicades a cadascuna d'aquestes tasques per part de l'estudiant (E) i els directors del projectes (DP).

Tasca	# tasca al Gantt	Data inici	Data fi	Dependències	Hores estimades E	Hores estimades DP
Introducció					**35**	**15**
Preparació del projecte	2	29/04/14	02/05/14		5	15
Lectura de documentació inicial	3	29/05/14	13/06/14	2	30	0
Introducció completa	4	13/06/14	13/06/14	3	0	0
Investigació API				4	**45**	**0**
Investigació API Facebook	6	23/06/14	18/07/14		15	0
Investigació API Twitter	7	16/06/14	15/07/14		15	0

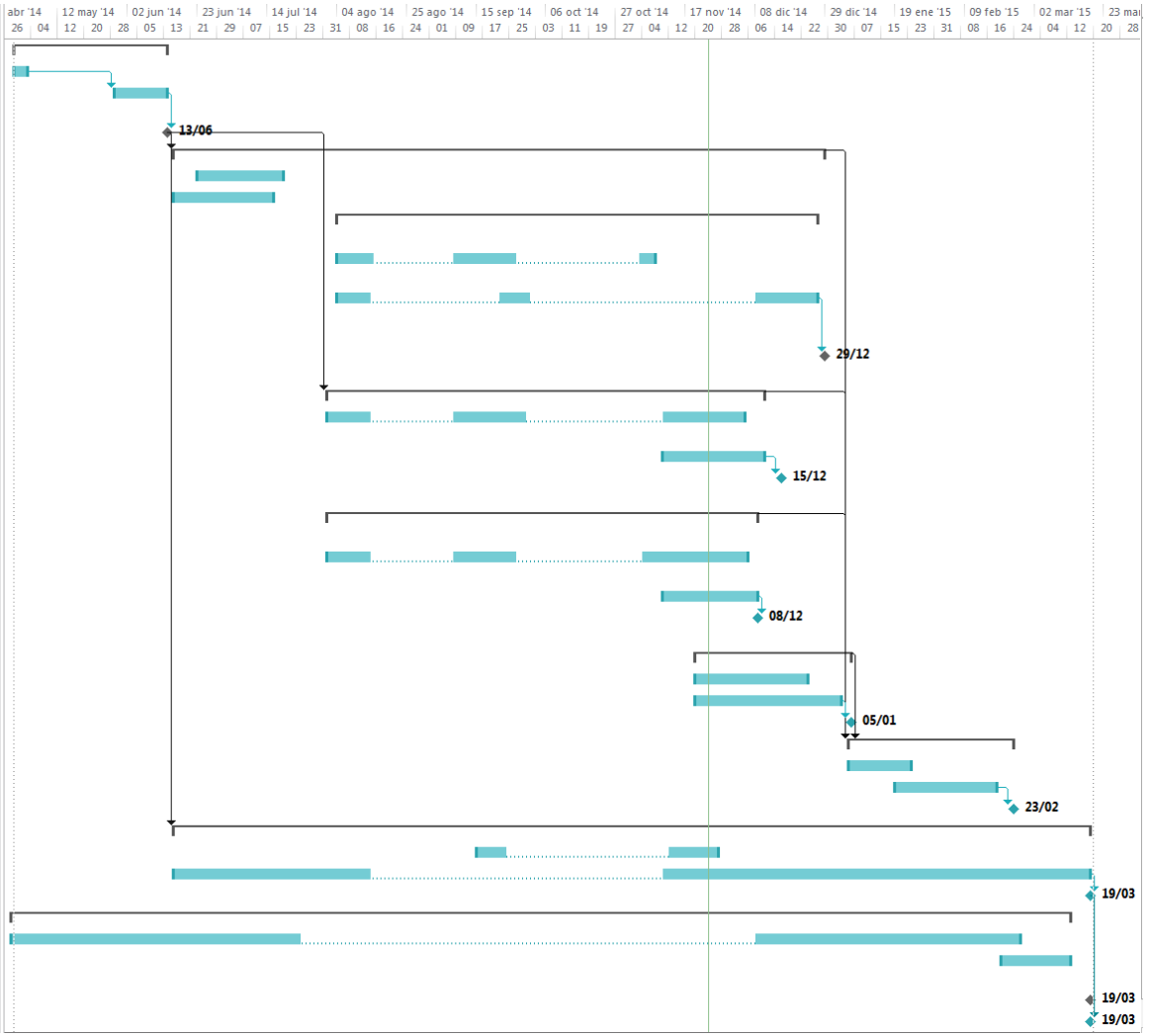
Investigació eines existents	8	04/08/14	07/11/14		15	0
Investigació APIs completada	9	07/11/14	07/11/14	6,7,8	0	0
Investigació *Machine Learning*				4	**40**	**0**
Investigació algoritmes existents	11	01/08/14	04/12/14		40	0
Investigació Machine Learning completada	12	04/12/14	04/12/14	11	0	0
Investigació *Sentiment Analysis*					**40**	**0**
Investigació algoritmes existents	14	01/08/14	05/12/14		40	0
Investigació Sentiment Analysis completada	15			14	0	0
Investigació WEKA					**55**	**0**
Aprenentatge funcionament WEKA	17	15/01/15	11/03/15		55	0
Investigació WEKA completada	18	11/03/15	11/03/15	17	0	0
Desenvolupament eines d'extracció i guardat de dades					**140**	**0**
Anàlisi	20	04/08/14	15/10/14		30	0
Disseny	21	20/10/14	23/12/14		30	0
Implementació	22	14/12/14	28/04/15		60	0
Proves	23	07/05/15	11/05/15		20	0
Desenvolupament finalitzat	24	11/05/15	11/05/15	23	0	0
Experimentació Machine Learning					**187**	**0**
Dataset	26	05/03/15	01/04/15		92	0
Captura de dades	27	05/03/15	09/03/15		10	0
Classificació	28	10/03/15	18/03/15		50	0
Neteja de dades	29	20/03/15	24/03/15		20	0
Creació del dataset d'entrenament	30	25/03/15	26/03/15		10	0
Creació del dataset de proves	31	01/04/15	01/04/15		2	0
Finalització creació datasets	32	01/04/15	01/04/15	31	0	0
Experimentació	33	01/04/15	24/04/15	32	95	0
Experimentació WEKA	34	13/04/15	20/04/15		55	0
Extracció de resultats	35	21/04/15	23/04/15	34	40	0
Experimentació completada	36	24/04/15	24/04/15	35	0	0
Finalització experimentació Machine Learning	37	24/04/15	24/04/15	33	0	0
Documentació				4	**185**	**0**
Preinforme	39	15/09/14	26/11/14		15	0
Memòria	40	16/06/14	22/09/16		170	0
Entrega del projecte	41	22/09/16	22/09/16	40	0	0

Control de projecte					**30**	**60**
Reunions de control	43	28/04/14	15/09/16		30	30
Revisió de la documentació	44	20/02/15	22/09/16		0	30
Defensa del projecte					**1**	**1**
Defensa del projecte	46	30/09/16	30/09/16	41	1	1
Projecte defensat	47	30/09/16	30/09/16	46	0	0
Total					**758**	**76**

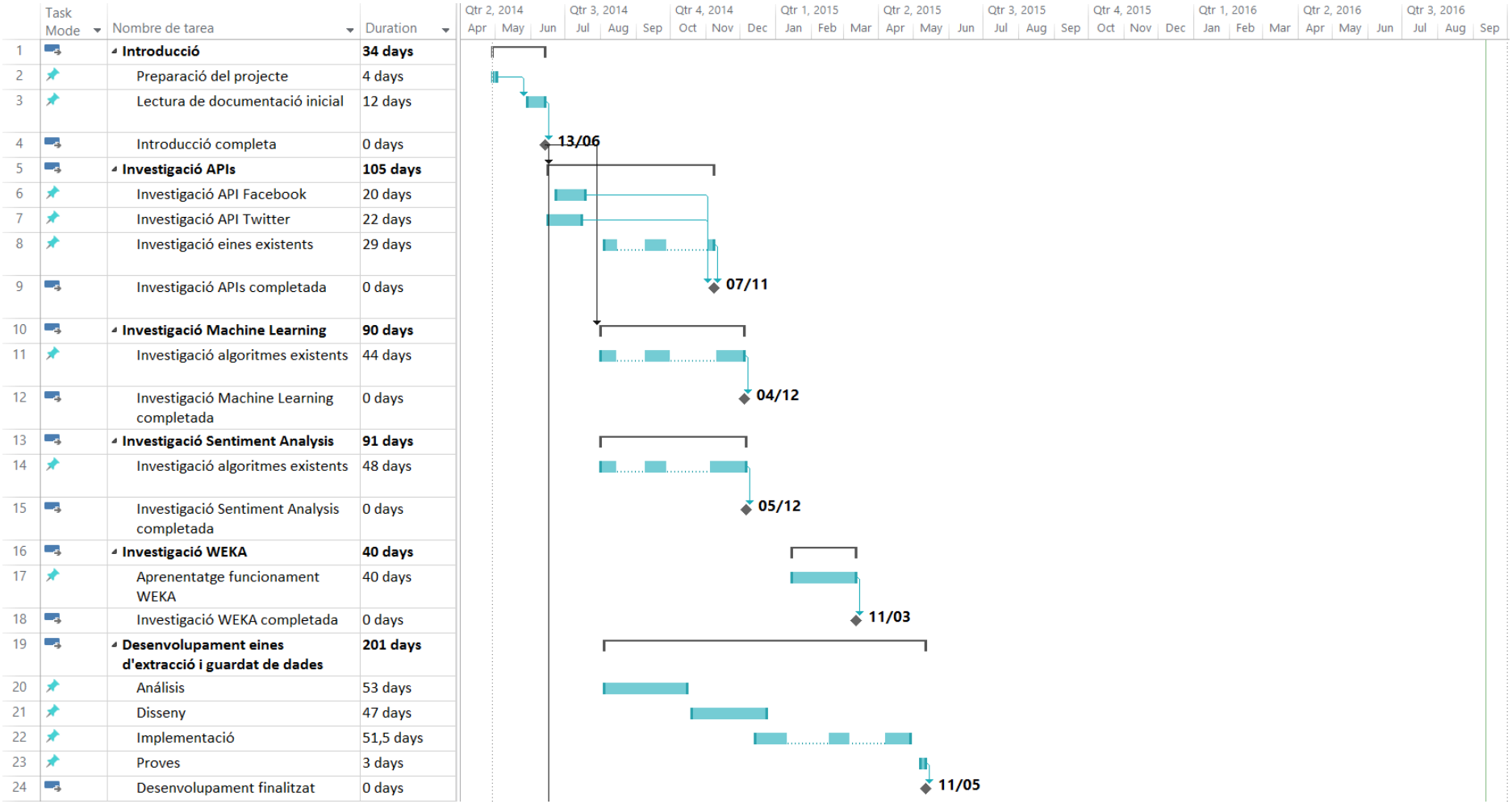
Taula 18. Planificació d'hores del projecte

Per finalitzar aquest apartat, es mostren els diagrames de Gantt corresponents a la planificació inicial i final del projecte. És important esmentar que les dates aparegudes en el diagrama de Gantt són utilitzades únicament com indicació de inici i finalització de les tasques, no com a indicador de l'esforç realitzat en cadascuna d'aquestes. Per tal de veure el conjunt d'hores utilitzat per cada tasca cal dirigir-se a la taula de planificació d'hores del projecte.

	Modo de	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1		**Introducció**	**34 días**	**mar 29/04/14**	**vie 13/06/14**	
2		Preparació del projecte	4 días	mar 29/04/14	vie 02/05/14	
3		Lectura de documentació inicial	12 días	jue 29/05/14	vie 13/06/14	2
4		Introducció completa	0 días	vie 13/06/14	vie 13/06/14	3
5		**Investigació APIs**	**140 días**	**lun 16/06/14**	**lun 29/12/14**	4
6		Investigació API Facebook	20 días	lun 23/06/14	vie 18/07/14	
7		Investigació API Twitter	22 días	lun 16/06/14	mar 15/07/14	
8		**Eines d'extracció i guardat de dades**	**105 días**	**lun 04/08/14**	**vie 26/12/14**	
9		Investigació eines existents	29 días	lun 04/08/14	vie 07/11/14	
10		Desenvolupament eines d'extracció i guardat de dades	30 días	lun 04/08/14	vie 26/12/14	
11		Investigació APIs completada	0 días	lun 29/12/14	lun 29/12/14	10
12		**Investigació Machine Learning**	**94 días**	**vie 01/08/14**	**mié 10/12/14**	4
13		Investigació algoritmes existents	44 días	vie 01/08/14	jue 04/12/14	
14		Implementació d'algoritmes	23 días	lun 10/11/14	mié 10/12/14	
15		Investigació Machine Learning completada	0 días	lun 15/12/14	lun 15/12/14	14
16		**Investigació Sentiment Analysis**	**92 días**	**vie 01/08/14**	**lun 08/12/14**	
17		Investigació algoritmes existents	48 días	vie 01/08/14	vie 05/12/14	
18		Implementació d'algoritmes	21 días	lun 10/11/14	lun 08/12/14	
19		Investigació Sentiment Analysis completada	0 días	lun 08/12/14	lun 08/12/14	18
20		**El·laboració marc de referència**	**33 días**	**jue 20/11/14**	**lun 05/01/15**	
21		Lectura de documentació	24 días	jue 20/11/14	mar 23/12/14	
22		Definició de l'estudi	32 días	jue 20/11/14	vie 02/01/15	
23		Marc de referència finalitzat	0 días	lun 05/01/15	lun 05/01/15	22
24		**El·laboració estudi**	**36 días**	**lun 05/01/15**	**lun 23/02/15**	**5;12;16;20**
25		Preparació de l'estudi	15 días	lun 05/01/15	vie 23/01/15	
26		Realització de l'estudi	23 días	lun 19/01/15	mié 18/02/15	
27		Estudi finalitzat	0 días	lun 23/02/15	lun 23/02/15	26
28		**Documentació**	**198 días**	**lun 16/06/14**	**jue 19/03/15**	**4**
29		Preinforme	18 días	lun 15/09/14	mié 26/11/14	
30		Memòria	136 días	lun 16/06/14	mié 18/03/15	
31		Entrega del projecte	0 días	jue 19/03/15	jue 19/03/15	30
32		**Control del projecte**	**229 días**	**lun 28/04/14**	**jue 12/03/15**	
33		Reunions de control	121 días	lun 28/04/14	mié 25/02/15	
34		Revisió de la memòria pels tutors	15 días	vie 20/02/15	jue 12/03/15	
35		**Defensa del projecte**	**0 días**	**jue 19/03/15**	**jue 19/03/15**	
36		Defensa del projecte	0 días	jue 19/03/15	jue 19/03/15	31

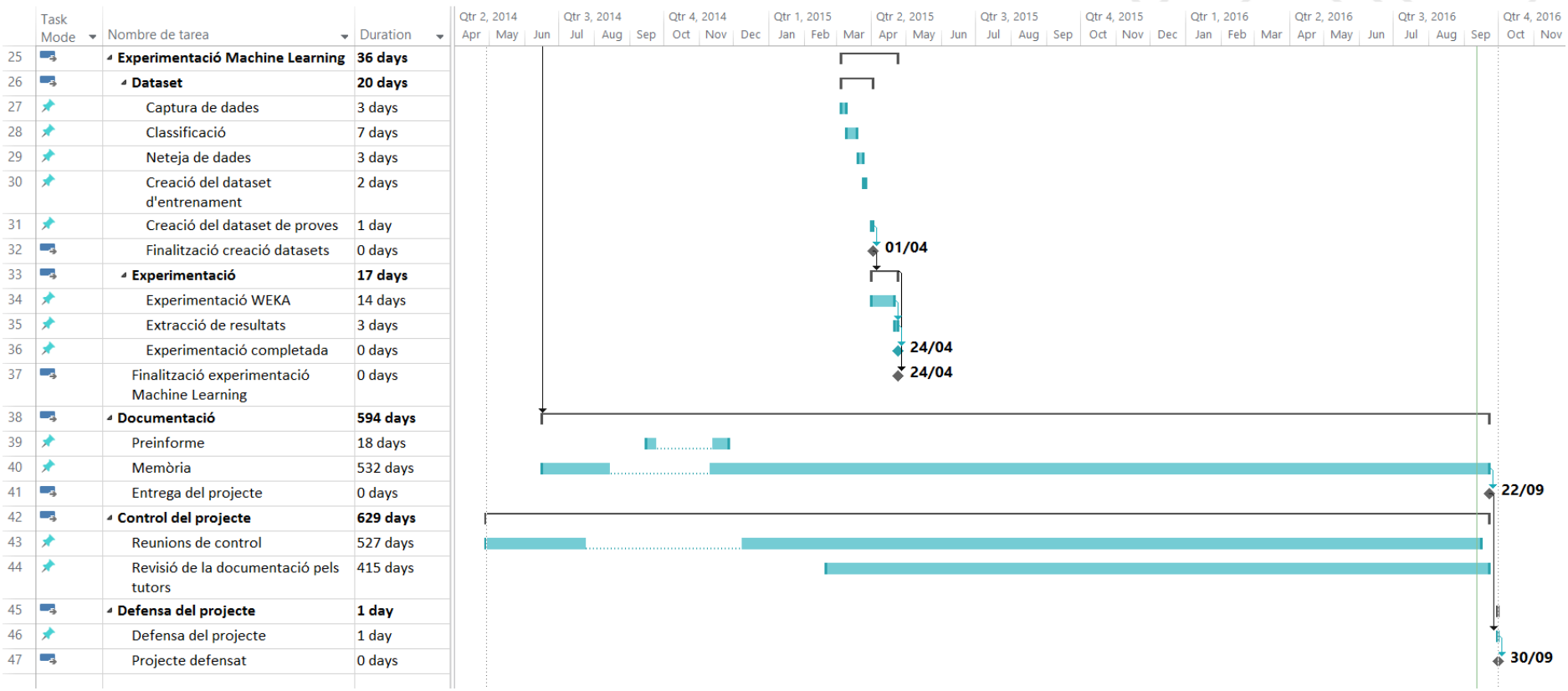


Imatge 34. Planificació inicial del projecte



	Task Mode	Nombre de tarea	Duration
1		**Introducció**	**34 days**
2		Preparació del projecte	4 days
3		Lectura de documentació inicial	12 days
4		Introducció completa	0 days
5		**Investigació APIs**	**105 days**
6		Investigació API Facebook	20 days
7		Investigació API Twitter	22 days
8		Investigació eines existents	29 days
9		Investigació APIs completada	0 days
10		**Investigació Machine Learning**	**90 days**
11		Investigació algoritmes existents	44 days
12		Investigació Machine Learning completada	0 days
13		**Investigació Sentiment Analysis**	**91 days**
14		Investigació algoritmes existents	48 days
15		Investigació Sentiment Analysis completada	0 days
16		**Investigació WEKA**	**40 days**
17		Aprenentatge funcionament WEKA	40 days
18		Investigació WEKA completada	0 days
19		**Desenvolupament eines d'extracció i guardat de dades**	**201 days**
20		Análisis	53 days
21		Disseny	47 days
22		Implementació	51,5 days
23		Proves	3 days
24		Desenvolupament finalitzat	0 days

Imatge 35. Planificació final del projecte (1)



	Task Mode	Nombre de tarea	Duration
25		**Experimentació Machine Learning**	**36 days**
26		**Dataset**	**20 days**
27		Captura de dades	3 days
28		Classificació	7 days
29		Neteja de dades	3 days
30		Creació del dataset d'entrenament	2 days
31		Creació del dataset de proves	1 day
32		Finalització creació datasets	0 days
33		**Experimentació**	**17 days**
34		Experimentació WEKA	14 days
35		Extracció de resultats	3 days
36		Experimentació completada	0 days
37		Finalització experimentació Machine Learning	0 days
38		**Documentació**	**594 days**
39		Preinforme	18 days
40		Memòria	532 days
41		Entrega del projecte	0 days
42		**Control del projecte**	**629 days**
43		Reunions de control	527 days
44		Revisió de la documentació pels tutors	415 days
45		**Defensa del projecte**	**1 day**
46		Defensa del projecte	1 day
47		Projecte defensat	0 days

Imatge 36. Planificació final del projecte (2)

4.2 Estimació econòmica

S'ha intentat estimar el cost del projecte de la millor manera tenint en compte la seva natura de projecte d'investigació difereix de les dels projectes clàssics de sistemes d'informació.

Tenint en compte aquestes circumstàncies, s'ha definit la següent situació:

- **Dos treballadors amb dos perfils diferents**: El primer treballador tindria el perfil de director, que assumiria el rol de cap de projecte. Aquest rol ha estat assumit per el Ferran Sabaté i l'Antonio Cañabate, els tutors d'aquest projecte. L'altre perfil, assumit per l'estudiant, tindria els rols de desenvolupador i data scientist.
- **Duració del projecte de cinc mesos**: Un treballador en jornades de 8 hores podria finalitzar el projecte amb una durada inferior a cinc mesos en jornada completa, tenint un petit marge per possibles contratemps que poguessin succeir.

En els següents apartats es desglossaran els costos humans, de software i ocupacionals que pot tenir el projecte tenint en compte aquests requisits.

4.2.1 Costos humans

A l'apartat anterior s'han esmentat els dos rols que treballaran en el projecte, a la taula 19 es definiran aquests rols, la funció que realitzaran i el cost d'aquests en el cas de que el projecte s'encarregués a una consulta..

Rol	Funció	Salari
Cap de projecte	Analitzar i dissenyar l'aplicació d'extracció i organitzar les tasques del projecte.	55 €/h[1]
Programador / Data scientist	Implementació i proves de l'aplicació de extracció de dades Realització d'experiments i extracció de resultats	33 €/h[1]

Taula 19. Rols del projecte

La taula 20 mostrarà les hores empleades per cadascun dels perfils empleats. Aquesta taula relaciona les tasques descrites en l'apartat anterior amb el perfil que realitza cadascuna d'elles.

Rol	Tasca	Hores empleades
Cap de projecte	Preparació del projecte	15
	Reunions de control	30
	Revisió de la documentació	30
	Assistència a la defensa del projecte	1
Total		**76**
Programador / Data scientist	Preparació del projecte	5
	Lectura de documentació inicial	30
	Investigació API Facebook	15
	Investigació API Twitter	15
	Investigació eines existents extracció de dades	15
	Investigació algoritmes existents de Machine Learning	40
	Investigació algoritmes existents de Sentiment	40

[1] Preus extrets de les tarifes d'e-tics

	Analysis	
	Aprenentatge funcionament WEKA	55
	Anàlisi de l'aplicació d'extracció de dades	30
	Disseny de l'aplicació d'extracció de dades	30
	Implementació de l'aplicació d'extracció de dades	60
	Proves de l'aplicació d'extracció de dades	20
	Captura de dades per l'experimentació	10
	Classificació de les dades extretes	50
	Neteja de dades de les dades extretes	20
	Creació del dataset d'entrenament	10
	Creació del dataset de proves	2
	Experimentació WEKA	55
	Extracció de conclusions dels resultats	40
	Elaboració del preinforme	15
	Elaboració de la memòria	170
	Reunions de control	30
	Defensa del projecte	1
Total		**758**

Taula 20. Relació dels perfils amb les tasques a realitzar

L'última taula d'aquest apartat mostra els costos humans del projecte un cop organitzades les tasques per perfil.

Rol	Hores empleades	Cost
Cap de projecte	76	4180 €
Programador / Data scientist	758	25014 €
Total		**29194 €**

Taula 21. Costos humans del projecte

4.3 Costos materials

En els costos informàtics es tenen en compte el cost dels ordinadors i les llicències del programari utilitzat en cas de que sigui necessari pagar-ne una. En el cas concret d'aquest projecte s'ha utilitzat un ordinador CORE i5 i llicències de Windows 8.1, Office 2013 i Project 2013.

El cost imputable de cada producte no és el preu original d'aquest, si no que se li ha aplicat un temps d'amortització dintre dels límits establerts pel BOE de novembre de 2014.

Material	Preu original	Unitats	Preu total	Termini amortització (mesos)	Cost imputable
Ordinador	650 €	2	1300 €	48	135,42 €
Llicència Windows 8.1	119 €	2	238 €	60	19,83 €
Llicència Office	119 €	2	238 €	60	19,83 €
Llicència Project	769 €	1	769 €	60	64,08 €
Total					**239,16 €**

Taula 22. Costos materials del projecte

4.4 Costos d'ocupació

Els costos d'ocupació d'aquest projecte s'han calculat suposant el lloguer d'una petita oficina moblada durant un parell de mesos pels dos treballadors.

Element	Cost parcial	Mesos	Cost
Oficina	500 €	5	2500 €
Connexió a internet	60 €	5	300 €
Total			**2800 €**

Taula 23. Costos d'ocupació del projecte

4.5 Costos totals

La taula 24 mostra els costos totals del projecte. S'ha aplicat de forma addicional unes despeses de contingència amb un valor del 15% del pressupost estimat. S'ha escollit aquest valor degut a l'alt risc que comporta la combinació de la realització d'un projecte d'aquest abast conjuntament amb un perfil de treballador amb nul·la experiència en l'àrea del Machine Learning.

Costos	Preu
Costos humans	29194 €
Costos materials	239,16 €
Costos d'ocupació	2800 €
Despeses contingència (15%)	4834,97 €
Total	37068,13 €

Taula 24. Costos totals del projecte

5 Extracció de dades de les API

En aquest apartat és procedirà a descriure el procés a seguir per tal de desenvolupar l'eina d'extracció de dades creada mitjançant les APIs de Twitter i Facebook.

Un cop realitzat l'estudi de les APIs de les xarxes socials i amb l'objectiu d'obtenir una eina per simplificar el procés d'obtenció de dades, els següents punts descriuran la especificació, el disseny i les decisions dutes a termes durant l'etapa d'implementació d'aquesta part del projecte.

En la part d'especificació es mostrarà un diagrama conceptual i la definició dels casos d'ús. A continuació, en el apartat de disseny es mostraran les decisions preses per tal de facilitar la implementació de l'aplicació. Finalment, en l'últim apartat referent a la implementació es mostraran al lector les decisions preses durant aquesta fase.

5.1 Especificació

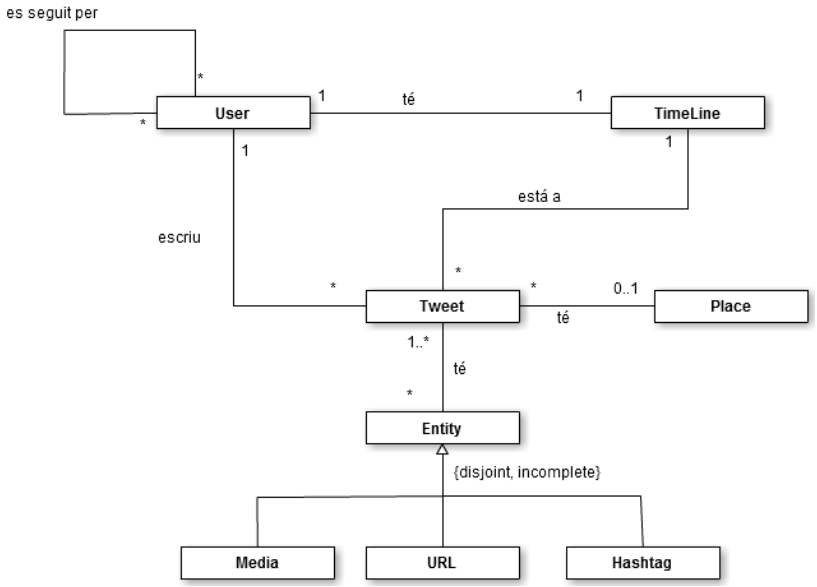
En aquest primer apartat es veuran els diagrames tant el conceptual, com els de casos d'ús dissenyats per tal de realitzar la especificació del projecte.

5.1.1 Diagrama conceptual

A continuació es mostren els diagrames conceptuals que seran utilitzats com a base de l'aplicació.

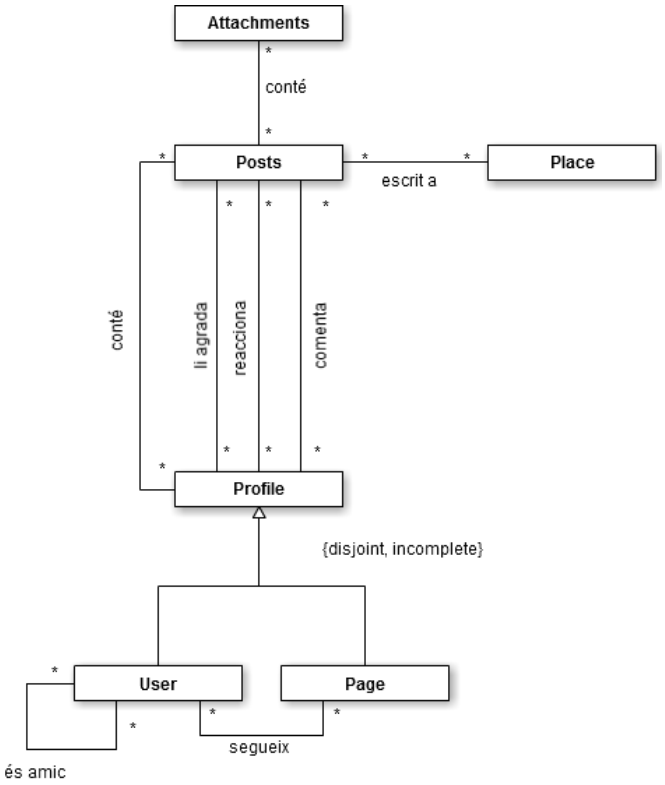
Aquests diagrames s'han separat en dos per tal de facilitar-ne la seva comprensió: per una banda el primer diagrama mostrarà l'esquema conceptual per la xarxa Twitter, mentre que el segon mostra el corresponent per Facebook.

5.1.1.1 Twitter



Imatge 37. Diagrama conceptual per la part de Twitter

5.1.1.2 Facebook



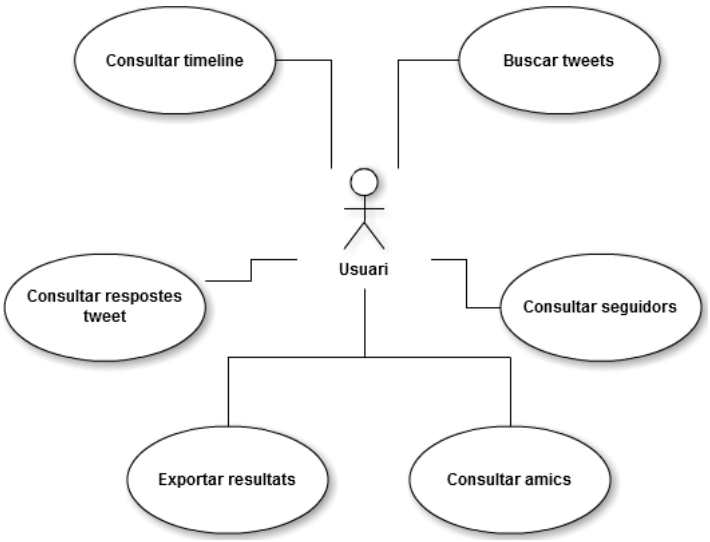
Imatge 38. Diagrama conceptual per la part de Facebook

5.1.2 Casos d'ús

Per tal d'obtenir les dades necessàries per fer estudis sobre l'àmbit de màrqueting digital, s'han definit una sèrie de casos d'ús que l'aplicació ha de complir per tal d'assolir l'objectiu. A continuació es descriuran aquestes funcions de forma similar a la realitzada en el cas anterior. Per una banda es mostraran el diagrama de casos d'ús per la xarxa Twitter, conjuntament amb una sèrie de taules per descriure aquests casos. Per altra banda en la segona part es fara la tasca homòloga per la xarxa Facebook.

5.1.2.1 Twitter

En aquest apartat es mostrarà al lector els casos d'ús detallats utilitzats en la xarxa Twitter.



Imatge 39. Diagrama de casos d'ús per la xarxa Twitter

Nom del cas d´ús	Consultar timeline
Autor	Usuari
Escenari principal d'èxit	• L'usuari accedeix a la opció de consultar el timeline d'un compte. • El sistema mostra el formulari per cercar el compte. • L'usuari introdueix les dades i pitja el botó de cerca. • El sistema mostra a l'usuari les dades del timeline demanat.

Taula 25. Cas d'ús de consultar timeline

Nom del cas d´ús	Buscar tweets
Autor	Usuari
Escenari principal d'èxit	• L'usuari accedeix a la opció de buscar tweets. • El sistema mostra el formulari de cerca de tweets. • L'usuari introdueix les dades de cerca. • El sistema mostra els tweets que corresponen als filtres utilitzats.

Taula 26. Cas d'ús de buscar tweets

Nom del cas d´ús	Consultar seguidors
Autor	Usuari
Escenari principal d'èxit	• L'usuari accedeix a la opció de consultar informació d'un usuari. • El sistema mostra el formulari de consulta d'informació d'un usuari. • L'usuari introdueix les dades de l'usuari del qual vol obtenir els seguidors. • El sistema mostra els seguidors de l'usuari corresponent.

Taula 27. Cas d'ús de consultar seguidors

Nom del cas d´ús	Consultar amics
Autor	Usuari
Escenari principal d'èxit	• L'usuari accedeix a la opció de consultar informació d'un usuari. • El sistema mostra el formulari de consulta d'informació d'un usuari. • L'usuari introdueix les dades de l'usuari del qual vol obtenir els amics. • El sistema mostra els amics de l'usuari corresponent.

Taula 28. Cas d'ús de consultar amics

Nom del cas d´ús	Consultar respostes tweet
Autor	Usuari
Escenari principal d'èxit	• L'usuari accedeix a la opció de consultar respostes d'un tweet. • El sistema mostra les respostes del tweet corresponent.

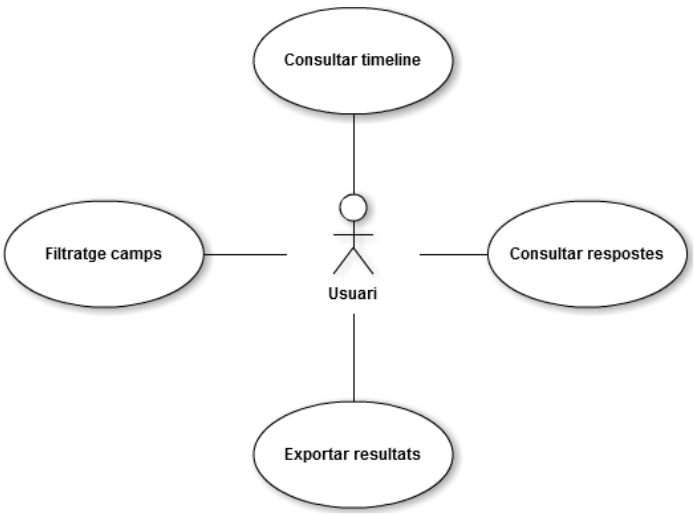
Taula 29. Cas d'ús de consultar respostes d'un tweet

Nom del cas d´ús	Exportar resultats
Autor	Usuari
Escenari principal d'èxit	• L'usuari executa l'opció d'exportació de resultats trobats en els casos d'ús anteriors. • El sistema exporta els resultats mostrats en format csv.

Taula 30. Cas d'ús d'exportar resultats

5.1.2.2 Facebook

Per finalitzar l'apartat de especificació, es mostraran el diagrama de casos d'ús per la xarxa de Facebook i les taules corresponents al detall d'aquests casos.



Imatge 40. Diagrama de casos d'ús de la xarxa Facebook

Nom del cas d´ús	Consultar timeline
Autor	Usuari
Escenari principal d'èxit	• L'usuari accedeix a la opció de consultar timeline de Facebook. • El sistema el formulari de selecció de perfil a consultar. • L'usuari introdueix les dades del perfil a consultar • El sistema mostra les dades del perfil.

Taula 31. Cas d'ús de consultar perfil

Nom del cas d´ús	Filtratge de camps
Autor	Usuari
Escenari principal d'èxit	• L'usuari accedeix a la opció de filtratge de camps d'un post. • El sistema el formulari de camps d'un post. • L'usuari introdueix els camps que vol visualitzar. • El sistema guarda les preferències de l'usuari.

Taula 32. Cas d'ús de consultar reaccions

Nom del cas d´ús	Consultar respostes
Autor	Usuari
Escenari principal d'èxit	• L'usuari accedeix a la opció de consultar timeline d'un usuari. • El sistema mostra el formulari de selecció de pàgina. • L'usuari introdueix les dades de la pàgina a consultar. • El sistema mostra les dades demanades. • L'usuari selecciona un post i utilitza la opció de consultar respostes. • El sistema retorna les respostes del post seleccionat.

Taula 33. Cas d'ús de consultar shares

Nom del cas d´ús	Exportar resultats
Autor	Usuari
Escenari principal d'èxit	• L'usuari executa l'opció d'exportació de resultats trobats en els casos d'ús anteriors. • El sistema exporta els resultats mostrats en format csv.

Taula 34. Cas d'ús d'exportar resultats

5.2 Disseny

Un cop definits els casos d'ús amb les accions que es duran a terme a l'aplicació, s'han pres les següents decisions referents al disseny.

5.2.1 Paradigma

S'ha utilitzat el paradigma de programació orientada a objectes per implementar la aplicació, aquest paradigma permetrà la reutilització de classes disminuint l'esforç necessari per desenvolupar el projecte. Addicionalment, gràcies a la facilitat de lectura que ofereix aquest

paradigma al permetre abstraure els conceptes de les classes pot facilitar la comprensió del codi a terceres persones.

5.2.2 Arquitectura

Per tal de desenvolupar l'aplicació, s'ha utilitzat una clàssica arquitectura de tres capes per tal d'afavorir la independència entre capes i poder realitzar canvis en cadascuna d'aquestes sense que aquests afectin a la resta del sistema.

La primera capa és la capa de presentació, en la qual hi haurà tot el conjunt de vistes per mostrar les funcionalitats i resultats de les accions de l'usuari.

La segona capa és la capa de negoci, aquesta capa conté tots els models de l'aplicació que representen tots els conceptes de l'esquema realitzat anteriorment a més de classes addicionals per tal de poder realitzar la lògica de negoci necessària per obtenir totes les funcionalitats.

La última capa és la capa de dades, tot i no haver base de dades en aquesta aplicació, aquesta capa correspon a l'accés a les APIs per part de l'aplicació.

5.2.3 Disseny de la capa de presentació

Per la capa de presentació s'ha definit la utilització del patró façana, amb el qual s'han pogut centralitzar totes les crides a la capa de domini amb una sola classe a fer servir de controlador.

Per implementar aquesta classe també es farà servir el patró Singleton, això és degut a que només es requereix una instància de la classe demanada, amb el qual realitzar instàncies addicionals seria una pèrdua de recursos no necessària.

5.2.4 Disseny de la capa de negoci

La capa de negoci conté al igual que la capa de presentació, un patró façana utilitzat per el controlador de domini. Aquest controlador realitzar la comunicació amb el controlador corresponent de la capa de presentació, obtenint d'aquesta forma un desacoblament entre les dues capes.

Addicionalment, s'ha procedit a la utilització de DTOs, o *Data Transfer Objects*, per tal de realitzar la comunicació amb la capa de presentació, ocultant d'aquesta forma la visibilitat de les classes del model a la capa superior.

5.2.5 Disseny de la capa de dades

Per la capa de dades no s'ha utilitzat cap tipus de patró significatiu degut a que l'accés a les dades es fa directament des d'una llibreria.

En aquesta capa, els objectes de la capa de negoci poden cridar directament als objectes d'aquesta, això és degut a la existència únicament de dos classes per aquesta. Una per accedir a l'API de Facebook i un altre per accedir a la de Twitter.

5.3 Implementació

Un cop preses les decisions de la etapa de disseny, és necessari decidir les tecnologies a utilitzar per tal de dur a terme la etapa d'implementació.

El llenguatge emprat per tal de realitzar la implementació ha sigut el llenguatge Java, llenguatge orientat a objectes en el qual ja s'havia adquirit experiència tan en l'àmbit acadèmic com en el professional, afavorint d'aquesta forma la rapidesa en el desenvolupament. Existeixen altres alternatives al llenguatge escollit, com C++ o C#. C++ és possiblement el llenguatge amb el qual els estudiants de la FIB tenen més experiència, tot i així, la seva dificultat no el feia aconsellable, doncs el temps emprat per resoldre problemàtiques del llenguatge no el feien adient per tal de ser utilitzat en una part del projecte en la qual l'esforç a dedicar havia de ser inferior a la part de Machine Learning d'aquest. Per altra banda, tot i tenir major experiència professional en C# i el framework .NET, al ser una tecnologia que en el moment present no és ensenyada en l'entorn acadèmic de la universitat, no era aconsellable ser utilitzada en cas de voler realitzar futures modificacions al codi.

Un cop decidit el llenguatge, la elecció de l'entorn de desenvolupament integrat, o IDE, va ser el següent pas a realitzar. En aquest cas la elecció estava entre dos aplicacions: Eclipse o NetBeans. La realització del projecte en un o altre IDE no té cap tipus de conseqüència futura, per aquest motiu s'ha utilitzat l'eina en la qual és té més experiència desenvolupant, Eclipse.

Amb el llenguatge i el programari a utilitzar escollits, les decisions preses a continuació afecten al desenvolupament directe de l'aplicació creada.

Decidir el mètode amb el qual es realitzen les crides a la API per part de l'aplicació ha sigut un pas important en el desenvolupament d'aquesta. En una primera versió es va optar per realitzar les crides a l'API directament mitjançant crides HTML i fitxes XML els quals contindrien la crida a realitzar i la configuració dels paràmetres a passar a les peticions a realitzar. En la versió final de l'aplicació aquesta metodologia s'ha descartat degut a la complexitat de la gestió dels arxius XML i la seva lectura i escriptura dinàmica per tal de realitzar les crides correctament. Per aquest motiu al final s'ha decidit la utilització de llibreries especialitzades per la comunicació a Twitter i Facebook com són Twitter4J i Facebook4J, dos de les llibreries més utilitzades actualment per tal de facilitar en gran mesura les tasques a realitzar.

Per finalitzar aquest apartat, es parlarà de les interfícies gràfiques utilitzades. Per poder millorar la usabilitat de l'aplicació era necessari realitzar una interfície gràfica d'usuari per aquesta. Degut a la nul·la experiència en la realització de interfícies per escriptori, aquest punt ha sigut un dels més incomodes en la seva implementació. Finalment s'ha decidit la utilització de Swing, una de les llibreries clàssiques de Java per la realització d'interfícies gràfiques, en detriment de SWT o JavaFX. La primera s'ha descartat per ser una llibreria coneguda pels seus múltiples errors, mentre que la segona, tot i ser la més actual i el futur estàndard en la programació Java, la documentació existent no és suficient per tal de realitzar un desenvolupament sense la necessitat de perdre temps cercant una documentació difícil de trobar.

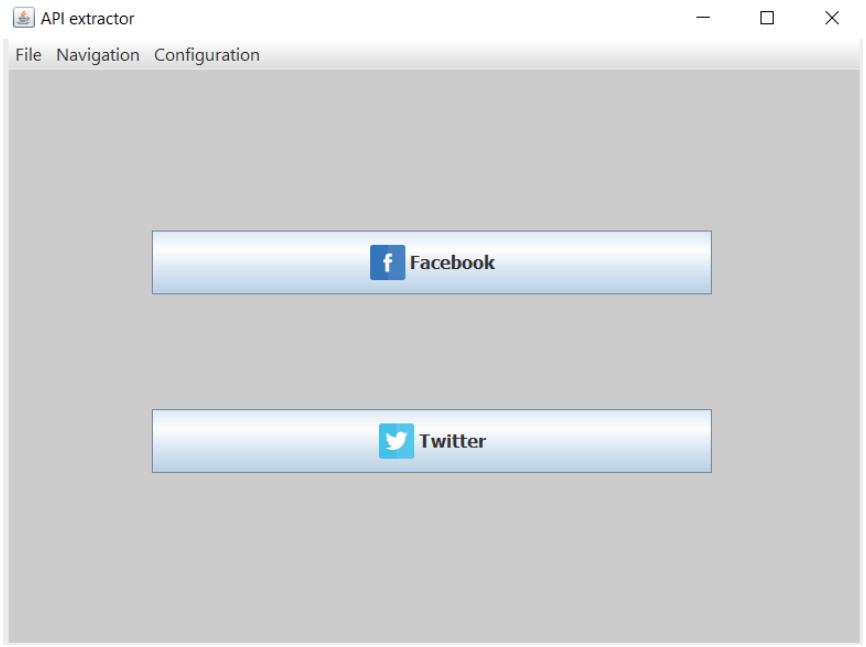
Un cop mostrada la informació sobre la implementació. Es mostrarà al lector un exemple de l'aplicació d'extracció de dades per tal de que comprovar el seu funcionament.

5.4 Exemple de captura de dades

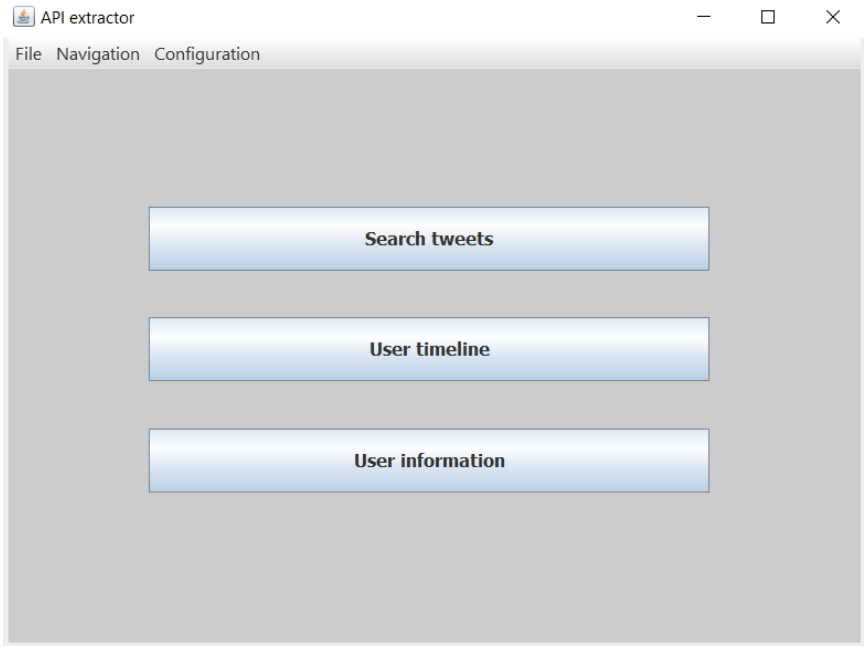
En el present apartat es mostrarà, a base de captures de pantalla, una de les funcionalitats disponibles en l'extractor de dades, concretament un exemple de l'extracció dels tweets del timeline de la FIB. El motiu de mostrar aquesta funcionalitat és degut a que aquesta, conjuntament amb la extracció de dades d'una pàgina de Facebook ha sigut una de les dos utilitzades per obtenir les dades necessàries per realitzar l'experimentació.

La pantalla principal de l'aplicació, mostrada a la imatge 41, permet a l'usuari seleccionar una de les dos xarxes disponibles per extreure dades.

Un cop seleccionada la opció de Twitter, es mostrarà per pantalla les opcions disponibles per consultar tal com es pot veure a la imatge 42. En aquest cas es disposen de tres opcions, la consulta de tweets diversos mitjançant la Search API, funcionalitat semblant a la disposada en les altres aplicacions analitzades, la consulta d'informació d'un usuari, on es podran obtenir informació diversa d'un usuari com els seguidors per exemple, i finalment la opció per la qual s'està realitzant l'exemple, la obtenció del timeline d'un usuari.

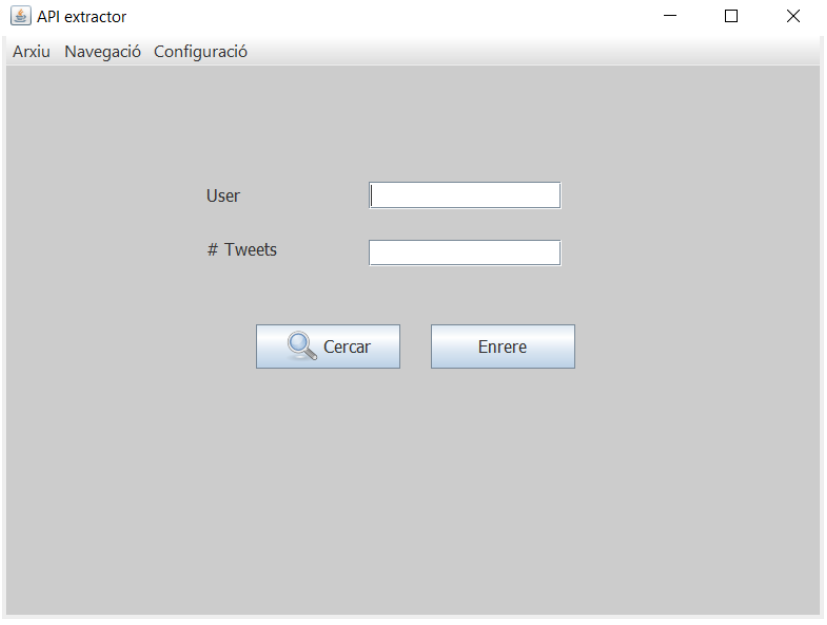


Imatge 41. Pantalla principal de l'aplicació



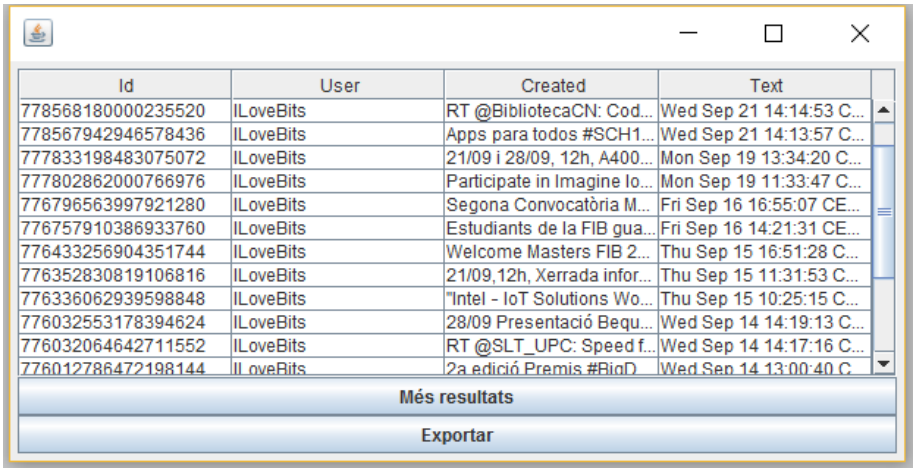
Imatge 42. Opcions disponibles per la xarxa Twitter

Un cop s'accedeix a la opció de consulta d'un timeline, l'aplicació mostrarà els camps que possa a disposició la API de Twitter per tal d'obtenir les dades. En el cas concret de les timelines, Twitter permet utilitzar com a paràmetres de la consulta l'usuari o el nombre de tweets que es volen recuperar. Tots aquests camps es poden veure a la imatge 43.



Imatge 43. Camps de consulta d'una timeline

En aquest cas concret, per il·lustrar l'exemple s'obtindran una sèrie de tweets del propi compte d'usuari de la FIB. Un cop escollides les opcions desitjades, l'aplicació realitzarà una cerca amb els paràmetres seleccionats.



Imatge 44. Pantalla de resultats

La imatge 44 mostra el resultat obtingut per l'aplicació. Les columnes mostrades són configurables des del menú de configuració de l'extractor. Addicionalment, utilitzant el botó dret del ratolí sobre alguna de les files, l'aplicació ofereix la funcionalitat addicional d'obtenir les respostes al tweet en qüestió en cas de que n'hi hagués. Ara bé, s'ha de tenir en compte que en el cas de Twitter, aquesta xarxa no ofereix la opció de recuperar tweets directament des de la seva API. Per aquest motiu, tot i que la problemàtica s'ha pogut sortejar, la implementació realitzada pot portar problemes si és utilitzada sobre timelines amb gran volum de dades.

Per tal de finalitzar l'exemple, tots els resultats obtinguts a la taula es poden exportar en format CSV mitjançant el botó corresponent. La imatge 45 mostra l'arxiu resultant d'aquesta exportació a partir dels resultats obtinguts.

	A	B	C	D	E
1	Id	User	Text	Date	Retweets
2	777833198483075000	ILoveBits	21/09 i 28/09, 12h, A4001, Xerra	Mon Sep 19 1	1
3	777802862000766000	ILoveBits	Participate in Imagine IoT #inno	Mon Sep 19 1	4
4	776796563997921000	ILoveBits	Segona Convocatòria Mobilitat	Fri Sep 16 16:	0
5	776757910386933000	ILoveBits	Estudiants de la FIB guanyen el c	Fri Sep 16 14:	4
6	776433256904351000	ILoveBits	Welcome Masters FIB 2016/201	Thu Sep 15 16	0
7	776352830819106000	ILoveBits	21/09,12h, Xerrada informativa	Thu Sep 15 11	1
8	776336062939598000	ILoveBits	Intel – IoT Solutions World Cong	Thu Sep 15 10	2
9	776032553178394000	ILoveBits	28/09 Presentació Beques Balse	Wed Sep 14 1	1
10	776032064642711000	ILoveBits	RT @SLT_UPC: Speed friending a	Wed Sep 14 1	6
11	776012786472198000	ILoveBits	2a edició Premis #BigData Talen	Wed Sep 14 1	1
12	775974580129304000	ILoveBits	Award ceremony for graduates	Wed Sep 14 1	1
13	775332105710690000	ILoveBits	27/09 Hack Day Roche, 21/09 de	Mon Sep 12 1	0
14	775279120381075000	ILoveBits	Presenta't com a voluntari a l	Mon Sep 12 1	1
15	774226738465349000	ILoveBits	Benvinguda als estudiants d'inte	Fri Sep 09 14:	1
16	774191511051333000	ILoveBits	Arriba la segona edició de @hac	Fri Sep 09 12:	4
17	773492213930201000	ILoveBits	CodeClub a les Biblioteques: en	Wed Sep 07 1	0
18	773136088890273000	ILoveBits	Benvinguda a nous estudiants al	Tue Sep 06 14	0
19	772727862612852000	ILoveBits	Premi 12x12 Dona TIC 2016 - 12	Mon Sep 05 1	0
20	771758968465190000	ILoveBits	RT @bibliotecnica: Tot sobre el	Fri Sep 02 19:	1
21	771258515075674000	ILoveBits	Bon dia!! Comencem curs 2016/	Thu Sep 01 10	2

Imatge 45. Arxiu obtingut a partir de l'exportació

5.5 Conclusions

Per concloure, tot i ja haver-se esmentat en altres punts, no deix de sorprendre la diferència de possibilitats que ofereixen les dues APIs. Mentre que l'API de Twitter sembla orientar-se cada cop més als científics que volen realitzar estudis amb les dades ofertes per la xarxa, Facebook en canvi limita cada cop més la informació a la qual es pot accedir. Aquesta actitud també s'ha pogut veure a l'hora de cercar informació per resoldre els problemes d'accessos a les dades que han anat sorgint durant la implementació. Mentre que en Twitter existeix una gran quantitat d'informació en forma de preguntes i respostes en diverses pàgines, la quantitat de consultes realitzades utilitzant l'API Graph de Facebook sembla ser bastant inferior. Tot i així, ambdós APIs tenen carències bastant importants. Actualment, la API de Twitter no ofereix de forma nativa la possibilitat de obtenir les respostes d'un tweet. Aquest fet, a l'hora de realitzar possibles estudis pot arribar a limitar la informació a obtenir. Per altra banda, la Graph API de Facebook tal com ja s'ha comentat amb anterioritat, ja no ofereix la possibilitat de accedir a dades públiques dels perfils d'usuaris a no ser que aquests hagin autoritzat a la aplicació per fer-ho, amb el qual, tota possibilitat d'extracció d'informació d'aquesta xarxa es limita a la extracció d'informació de les pàgines d'empreses.

6 Classificació de textos

El present apartat mostrarà la metodologia utilitzada per realitzar l'experimentació utilitzant l'aprenentatge automàtic.

En el primer subapartat es descriurà al lector el mètode d'obtenció dels diferents datasets utilitzats per l'experimentació següents apartats es mostrarà la experimentació completa utilitzant l'aprenentatge automàtic, es començarà explicant l'obtenció dels diferents datasets utilitzats per realitzar l'experimentació. El segon mostrarà el procediment dut a terme per tal d'obtenir els resultats, mentre que en el tercer punt es podran veure els resultats obtinguts. Finalment, s'exposaran en l'últim apartat les conclusions extretes a partir dels resultats de l'experimentació.

6.1 Obtenció dels corpus

6.1.1 Introducció

El primer pas abans de realitzar cap tipus d'experimentació era l'obtenció d'un conjunt de dades per tal de poder entrenar els algoritmes a utilitzar posteriorment. Per aquest motiu es va procedir a l'obtenció de tres datasets diferenciats per tal d'obtenir aquestes dades.

El primer dataset, com s'ha esmentat en apartats anteriors, va ser proporcionat per la doctora Irena Pletikosa, gràcies a la directa intervenció del codirector del projecte, Antonio Cañabate. La doctora especialista en l'anàlisi de les xarxes socials, va utilitzar el dataset en el seu estudi *Online Engagement Factors on Facebook Brand Pages.*

En l'article esmentat, es realitza una classificació de 612 textos extrets de la xarxa en diverses categories com són post de tipus afecte, compartir informació, peticions d'aquesta o suggeriments.

El segon i tercer datasets, van ser creats per tres motius principalment:

- **Quantitat de dades**: La poca quantitat de dades ofertes pel dataset de Pletikosa augmentava la possibilitat de que les dades obtingudes de l'experimentació fossin poc creïbles, amb el qual era necessari obtenir un subconjunt major per millorar les possibilitats d'èxit de l'experimentació.
- **Categorització**: Durant la realització del preinforme del projecte es van definir tres categories de post a les xarxes sobre les quals realitzar un possible estudi de màrqueting digital. Aquestes categories, definides posteriorment, no corresponen a les ofertes per Pletikosa amb el qual per la realització de l'experimentació s'han volgut utilitzar textos que puguin ser classificats en aquestes.
- **Incorporació del sector com informació d'entrada**: El director i codirector del projecte, a causa de la naturalesa dels seus estudis anteriors, estaven especialment interessats en conèixer l'efecte que produiria en la bonança dels algoritmes la incorporació del sector de negoci al qual pertanyia la marca. Per aquest motiu, es va procedir a incorporar el sector al qual pertanyien les marques en el tercer dataset utilitzat, sent aquesta l'única diferència respecte al segon, el qual no conté aquest tipus d'informació.

Com ja s'ha esmentat en els punts anteriors, les instàncies del segon i tercer dataset estan classificades en tres categories diferents, aquestes estan definides a continuació:

- **Remuneració**: Els posts dintre d'aquesta categoria ofereixen algun tipus de remuneració econòmica a l'usuari, sigui en forma de concurs patrocinat per l'empresa o bé alguna classe d'oferta.
- **Informatiu**: Aquesta categoria fa referència a aquelles comunicacions en les quals els Community Managers ofereixen informació sobre la pròpia empresa o sobre algun llançament d'algun nou producte o servei d'aquesta.
- **Entreteniment:** La última categoria defineix aquells posts escrits que no tenen una relació directa amb la empresa i tinguin la intenció d'entretenir a l'usuari. En aquesta categoria s'hi poden incloure des dels comentaris sobre un partit de futbol, fins a preguntar quins són els plans dels usuaris pel cap de setmana.

La taula 35 mostra un resum dels tres datasets d'entrenament utilitzats incloent la nomenclatura a partir de la qual es farà referència a aquests per tal de facilitar la lectura del document.

Nomenclatura	Contingut	Classes
Dataset A	Dataset proporcionat per Pletikosa	• Afecte • Compartir informació • Peticions informació • Suggeriments
Dataset B	Dataset creat utilitzant instàncies extretes de les xarxes socials utilitzant l'extractor desenvolupat.	• Informació • Entreteniment • Retribució
Dataset C	Dataset amb les mateixes dades que el dataset B, amb la diferència de tenir els sectors incorporats en la informació de les instàncies.	• Informació • Entreteniment • Retribució

Taula 35. Resum dels datasets d'entrenament creats per l'experimentació

6.1.2 Factors de decisió en la creació dels nous datasets

Un cop decidida la creació dels datasets amb les categories anteriors, s'han tingut presents una sèrie de factors per portar a terme la seva elaboració, per exemple: la quantitat de post a extreure, els sectors a analitzar i finalment les marques pertanyents a aquests.

En el moment de decidir la quantitat de post a extreure de les xarxes socials s'havia de tenir en compte principalment el factor temps. Tot i que l'extracció dels post hagi sigut relativament ràpida gràcies a l'extractor desenvolupat, la posterior neteja i categorització requeria d'un esforç temporal bastant més alt. Per aquest motiu, tenint en compte aquest fet es va decidir realitzar una extracció de 1000 post per tal de servir d'entrenament dels algoritmes i 200 post addicionals per tal d'executar les proves del model creat després de realitzar l'entrenament. De tots aquests posts extrets, aproximadament un 50% pertany a la xarxa social Facebook, mentre que l'altre 50% pertany a la xarxa Twitter, això s'ha fet així per intentar minimitzar el possible impacte que pugui tenir el medi del qual s'extrauen les dades sobre la seva classificació, tot i que no s'han trobat estudis que demostrin una relació directa entre les diferències de tipus de contingut creat entre les xarxes Twitter i Facebook.

Pel que fa als sectors a analitzar, s'han escollit aquells que poguessin tenir relació amb les categories definides anteriorment, per exemple, és evident que el sector de venta online té

més possibilitats d'incloure post de tipus de remuneració a la pàgina de la seva marca que no pas el sector d'automobilisme.

Un cop escollida la quantitat de post i sectors a abastar, s'han agafat les dades de les marques més conegudes o bé amb més presència a la xarxa social, per aquest motiu, pàgines o blogs diversos com TrueSocialMetrics, entre altres, han ajudat a trobar aquestes marques.

Les conclusions obtingudes d'aquestes eleccions es poden veure a les taules 39 i 40. Finalment, en els datasets B i C es disposen de quatre sectors i un mínim de tres marques per cadascun d'aquests. El motiu pel qual el nombre de marques no és equivalent en tots els sectors, va ser únicament degut a la problemàtica d'extracció de dades filtrades a Twitter. El comportament d'aquesta xarxa és poc idoni per extreure dades de pàgines de les marques amb gran interacció amb els seus usuaris, ja que quan es realitza la crida per obtindré els tweets del seu timeline, Twitter no només retorna els tweets propis, sinó que també retorna les converses de la marca amb altres usuaris. Aquest fet provoca que s'haurien de realitzar un gran nombre de crides a la API per tal d'aïllar els tweets de comunicació general de les marques de les converses d'aquestes amb altres usuaris de la xarxa. Per tant, per estalviar temps i davant de la complicació de realitzar múltiples crides en la versió preliminar de l'extractor, es va decidir extreure les dades d'un altre marca.

Per finalitzar aquest apartat, es mostren les taules 36 i 37. La primera taula mostra els sectors i marques escollides de les quals s'han extret les instàncies per incloure en els datasets B i C. És important recordar en aquest punt, que tot i que els dos datasets contenen les mateixes instàncies, la informació del sector com atribut únicament apareix en el dataset C tal com es pot veure en la taula 35 de l'apartat anterior.

Xarxa	Sector	Marca
Twitter	Viatge	eDreams
		LastMinute
		Tripadvisor
	Beguda	Budweiser
		Pepsi
		Redbull
		MonsterEnergy
Facebook	Menjar	Dominos Pizza
		Boston Pizza
		Little caesars
	Venta online	Amazon
		Zavvi
		Thehut

Taula 36. Relació de xarxes, sectors i marques escollides pel dataset d'entrenament B i C

La taula 37 mostra la mateixa relació però a diferència de l'anterior, la informació mostrada pertany als datasets de 200 instàncies de proves utilitzats per validar els resultats obtinguts de l'entrenament mitjançant els datasets B i C. Anàlogament al cas anterior, les proves per validar el dataset C contindran la informació del sector, a diferència del conjunt d'instàncies per validar el dataset B, que tot i ser les mateixes, la informació del sector al qual pertanyen aquestes no serà present.

Xarxa	Sector	Marca
Twitter	Viatge	Expedia
	Menjar	PizzaExpress
Facebook	Beguda	Dr.Pepper
	Venta online	RakutenUK

Taula 37. Relació de xarxes, sectors i marques escollides pel dataset d'entrenament

6.1.3 Categorització de les instàncies

El nombre d'instàncies extretes per classificar en els datasets B i C, ha sigut bastant superior a les 1200 finals, això és degut a que les primeres classificacions proveïen un desbalanceig important entre categories. Un desbalanceig superior al 80% entre les classes, podria provocar desajustos importants en l'entrenament dels algoritmes (Gomez Hidalgo et al., 2013), per aquest motiu, tot i existir tècniques com el *up-sampling* o el *down-sampling* (Provost et al., 2000) s'ha preferit per motius d'eficiència temporal, classificar un nombre superior d'instàncies i eliminar les sobrants. Això ha permès anivellar el pes entre categories finals evitant el desajustos esmentats anteriorment.

Per altra banda, les instàncies del dataset A ja estaven classificades des d'un inici en diverses categories. D'aquestes se n'han escollit les quatre que contenien un volum d'instàncies suficientment elevat per tal de no produir cap tipus de desbalanceig en l'entrenament dels algoritmes. Aquesta decisió ha sigut motivada degut a que alguna de les categories restants no contenien tan sols un 4% del total d'instàncies contingudes en el total del dataset. Així doncs, amb aquesta feina ja realitzada, l'única tasca necessària ha sigut separar les 507 instàncies restants en un dataset d'entrenament i un de proves, tenint cadascun d'ells 400 i 107 instàncies respectivament.

Per concloure aquest apartat, les taules 38, 39, 40 i 41 mostren al lector el nombre d'instàncies per categoria en els datasets que s'utilitzaran posteriorment per l'experimentació.

Categoria	Instàncies	Percentatge (%)
Informació	311	31.1
Entreteniment	475	47.5
Retribució	214	21.4
Total	**1000**	**100**

Taula 38. Classificació d'instàncies dels datasets d'entrenament B i C

Categoria	Instàncies	Percentatge (%)
Informació	88	44
Entreteniment	51	25.5
Retribució	61	30.5
Total	**200**	**100**

Taula 39. Classificació d'instàncies del dataset de validació dels datasets B i C

Categoria	Instàncies	Percentatge (%)
Afecte	116	29
Petició d'informació	57	14.25
Suggeriments	108	27
Compartir	119	29.75
Total	**400**	**100**

Taula 40. Classificació d'instàncies del dataset d'entrenament de Pletikosa

Categoria	Instàncies	Percentatge (%)
Afecte	21	19.63
Petició d'informació	35	32.71
Suggerencies	33	30.84
Compartir	18	16.82
Total	**107**	**100**

Taula 41. Classificació d'instàncies del dataset de proves de Pletikosa

6.1.4 Neteja i preparació dels datasets

Un cop totes les instàncies han sigut classificades, es necessari realitzar una neteja dels datasets per tal de poder realitzar l'entrenament posterior. L'objectiu d'aquesta és eliminar elements superficials que no aporten informació útil als algoritmes per realitzar la classificació. En el cas del projecte en qüestió, la neteja és encara més necessària, doncs l'àmbit del qual s'han extret les instàncies és propici a incloure aquesta informació supèrflua en forma de *hashtags*, *smileys*, entre altres.

Addicionalment, un cop s'ha realitzat la neteja dels datasets, s'han de preparar aquests per tal de poder carregar la seva informació al WEKA ja que alguns símbols poden ser llegits de forma incorrecta.

A continuació, s'enumeren els passos que s'han realitzat, en ordre, per tal de netejar el dataset:

- **Eliminació de símbols**: S'han eliminat símbols dels textos retornats per les APIs, com poden ser *\n*, *&gt*, entre altres...
- **Eliminació dels enllaços:** Una gran quantitat d'instàncies extretes, sobretot aquelles instàncies extretes de Twitter, contenien diverses URL amb diverses finalitats, per exemple la direcció d'una foto, d'un tweet, de la pàgina d'un blog, etc...
- **Eliminació de hashtags:** Sobretot a Twitter, moltes instàncies incorporen una paraula amb un símbol de # davant, anomenada hashtag. Habitualment no aporten un valor addicional a la frase, per aquest motiu només s'han conservat aquells que si que proporcionen algun tipus de valor. L'exemple 3 de la taula 42 en mostra un exemple d'aquest cas.
- **Eliminació de símbols #:** Un cop s'han eliminat els hashtags superflus, s'ha eliminat el símbol '#' d'aquells que si que proporcionen algun tipus d'informació addicional.
- **Substitució de ' per \':** WEKA pot tenir problemes a l'hora de llegir frases amb apòstrof, per aquest motiu, ha sigut necessari escapar aquests per tal de que el programa en faci una correcta lectura.

La taula 42 mostra fins a tres exemples dels passos realitzats.

Exemple 1	
Original	Top 6 unspoiled places in Italy http://ow.ly/LwJtW #travel #traveltips
Resultat	Top 6 unspoiled places in Italy
Exemple 2	
Original	Waking up to this view... ? \nLet's be adventurers! ? http://t.co/GvhxbJURJ1
Resultat	Waking up to this view... ? Let\'s be adventurers! ?
Exemple 3	
Original	Here's why you should go on holiday to the #Bahamas! http://ow.ly/KSnz6 #travel
Resultat	Here\'s why you should go on holiday to the Bahamas!

Taula 42. Exemples de la neteja i preparació realitzada

6.2 Experimentació

6.2.1 Objectius de l'experimentació

Tal com s'ha anat esmentat al llarg d'aquest document, el principal objectiu del projecte és conèixer la viabilitat de la utilització d'eines de Machine Learning per ajudar a la investigació de les característiques que tenen les publicacions de les marques en relació als seus seguidors a les xarxes socials.

Tenint en ment l'objectiu anterior i amb la investigació realitzada sobre Machine Learning d'aprenentatge supervisat, s'han decidit dos problemàtiques a resoldre per tal d'analitzar la possible viabilitat:

- **Longitud dels datasets**: La primera de les dos preguntes de les quals aquest projecte intentar trobar resposta, consisteix en esbrinar si existeix un punt en els quals el nombre d'instàncies passades als algoritmes de Machine Learning deix d'afectar a la eficiència d'aquests en el moment de classificar instàncies. La importància d'aquesta pregunta radica en el fet de saber quin és el nombre mínim d'instàncies necessàries per construir un dataset nou sobre el qual fer un estudi.
- **Encert dels algoritmes**: La segona problemàtica existent i la més important a resoldre en aquest projecte, consisteix en conèixer si els algoritmes utilitzats poden classificar noves instàncies amb un percentatge alt d'encert. Respondre afirmativament a aquesta pregunta suposaria la possibilitat d'incorporar l'aprenentatge automàtic en futurs estudis d'investigació relacionats en l'àmbit de l'anàlisis de les comunicacions de les marques en les xarxes socials.

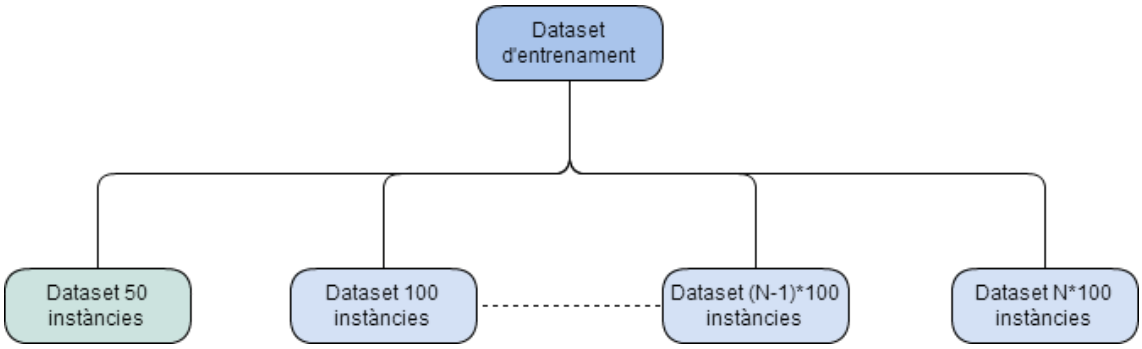
6.2.2 Metodologia de proves

Per tal de complir els objectius del punt anterior, s'han seguit una sèrie de passos per realitzar els anàlisis en qüestió.

6.2.2.1 Divisió dels datasets

El primer pas a realitzar ha consistit en dividir els datasets d'entrenament creats anteriorment en una sèrie de conjunts de mida inferior i incremental respecte uns als altres, cada subconjunt creat ha afegit un total de 100 instàncies noves respecte al conjunt anterior, excepte els dos subconjunts més petits, que tenen una mida de 50 i 100 instàncies respectivament. L'objectiu a perseguir amb aquest pas és comprovar l'existència del punt a partir del qual l'addició d'instàncies al dataset d'entrenament no afecta al percentatge d'encert que té aquest a l'hora de classificar noves instàncies. Aquest pas afecta a tots els datasets d'entrenament utilitzats durant la realització de l'experimentació, no és així en els de proves, doncs no té cap sentit

dividir el nombre d'instàncies noves a classificar per comprovar l'eficiència dels algoritmes. La imatge 46 mostra de forma visual la tasca realitzada.



Imatge 46. Divisió dels datasets en conjunts de mida inferior

6.2.2.2 Transformació a format ARFF

A continuació, un cop s'han obtingut tots els subconjunts desitjats, s'ha procedit a passar cadascun dels datasets obtinguts junt amb els de proves al format ARFF, el conjunt de dades acceptat per WEKA i definit a l'apartat corresponent de l'estat de l'art, amb l'objectiu d'utilitzar-los com entrada per aquesta aplicació. Per realitzar aquesta transformació s'han utilitzat diversos atributs. Pel dataset A se n'han definit dos, el primer, de tipus string, ha sigut utilitzat per tal de definir el contingut del missatge el qual serà classificat posteriorment, mentre que el segon, que correspon a la classificació donada al missatge, serà de classe nominal. Els mateixos atributs han sigut definits pel dataset B, el dataset creat per la experimentació que no conté informació dels sectors, amb la única diferència del canvi de categories corresponent. Finalment, el dataset C conté tres atributs, mentre que dos són idèntics al dataset B, text i category, el tercer atribut, també de tipus nominal, conté informació sobre el sector al que correspon al post.

Les taules 43, 44 i 45 mostren la relació entre atributs i datasets explicada anteriorment.

Atribut	Tipus	Contingut de l'atribut
Text	String	Comunicacions extretes per la doctora Pletikosa
Category	Nominal	• A (Affect) • I (Information Inquiry) • R (Request & Suggestions) • S (Sharing)

Taula 43 Atributs del dataset A.

Atribut	Tipus	Contingut de l'atribut
Text	String	Comunicacions extretes de les pàgines de les marques
Category	Nominal	• I (Information) • E (Entertaintment) • R (Retribution)

Taula 44. Atributs del dataset B

Atribut	Tipus	Contingut de l'atribut
Text	String	Comunicacions extretes de les pàgines de les marques
Category	Nominal	• I (Information) • E (Entertaintment) • R (Retribution)
Sector	Nominal	• Travel • Food • Drinks • OnlineShopping

Taula 45. Atributs del dataset C

6.2.2.3 Configuració de les proves

Un cop s'han obtingut els datasets en el corresponent format ARFF, ja s'ha pogut procedir a realitzar la experimentació del projecte.

Com ja s'ha comentat en apartats anteriors, molts algorismes de Machine Learning, requereixen de un preprocessament previ de les dades per tal de poder ser utilitzats. WEKA ofereix dos mètodes per tal de realitzar el preprocessament, o bé mitjançant la GUI, o bé mitjançant línia de comandes.

En aquest projecte s'ha decidit dur a terme el segon mètode, això és degut a que la realització d'aquest preprocessament per línia de comandes permet realitzar la transformació dels arxius ARFF inicials de forma pràcticament automàtica amb un sol script de comandes.

Per tal de decidir els paràmetres adients, s'han provat una sèrie de configuracions per tal d'obtenir la millor combinació de paràmetres per tal d'augmentar el rendiment dels algorismes. En aquest punt, es van fer les proves de configuracions mitjançant la tècnica de validació creuada, tècnica que ha proporcionat bons resultats en els datasets emprat. Malauradament, aquesta metodologia va evitar observar el comportament dels algorismes amb el dataset de proves corresponent. Tot i així, en les taules de resultats de les configuracions, es mostra el resultat de la validació mitjançant aquest últim a forma de comparativa amb la validació creuada.

Els següents subapartats d'aquest document, mostren al lector la configuració general de les proves i cadascuna de les configuracions realitzades.

6.2.2.3.1 Procés de configuració

Cada una de les configuracions consisteix en l'execució de dos processos, el de transformació a vector de paraules, conegut a WEKA com StringToWordVector i el procés de selecció d'atributs, conegut com AttributeSelection. Addicionalment, totes les configuracions tenen en comú els següents punts:

- **Algorisme de classificació**: Totes les configuracions utilitzen SVM per tal de realitzar les proves. Aquest algoritme com ja s'ha esmentat en punts anteriors és dels algorismes més utilitzats en la realització d'estudis de recerca).
- **Selector de selecció d'atributs**: Per tal de realitzar la selecció d'atributs, s'ha seleccionat el mètode del guany d'informació, al ser aquest un dels selectors amb millors resultats (Joachims et al,. 1997).

- **Batch-filtering**: WEKA no pot processar un dataset d'entrenament i un dataset de proves si aquests dos no comparteixen els mateixos atributs, per aquest motiu és necessari aplicar el paràmetre de batch-filtering en els processos a realitzar per tal de que els datasets a utilitzar tinguin el mateix conjunt d'atributs. En conclusió, caldrà aplicar aquest paràmetre en la transformació de cadascun dels datasets d'entrenament conjuntament amb el dataset de proves per tal de que els subconjunts i els datasets de proves tinguin els mateixos atributs.
- **Dataset d'entrenament**: Per tal de realitzar les configuracions, s'ha utilitzat el dataset B amb el màxim d'instàncies possibles per trobar-ne el millor rendiment.

Un cop establerts els paràmetres comuns de les configuracions, ha sigut necessari definir les diferències entre cadascuna d'aquestes. Entre els dos processos a realitzar, existeixen una sèrie de paràmetres que poden modificar el rendiment dels algorismes els quals han sigut modificats en cadascuna de les configuracions. Els utilitzats per diferenciar es poden trobar a la taula 46:

StringToWordVector	AttributeSelection
Stemmer Transformació TFIDF/Freqüència de paraules StopWords Tokenizer	Llindar de guany d'informació

Taula 46. Atributs rellevants en la configuració dels processos

Per finalitzar, en cadascun dels següents apartats es mostra al lector les diverses configuracions realitzades , oferint a l'últim la taula de resultats de cadascuna de les proves realitzades.

6.2.2.3.2 Configuració 1

La primera configuració s'ha realitzat utilitzant una sèrie de paràmetres recomanables en la majoria d'estudis, realitzant la transformació TFIDF, utilitzant un stemmer, un separador de paraules bàsic, un diccionari d'stopwords ofert per WEKA i finalment, eliminar aquells atributs que no proporcionen cap guany de valor.

Filtre	Paràmetre	Valor
StringToWordVector	Stemmer	LovinsStemmer
	Tokenizer	WordTokenizer
	TFIDF/Freqüència de paraules	TFIDF
	StopWords	Rainbow
AttributeSelection	InfoGainAttributeEval	>0

Taula 47. Configuració de proves 1

6.2.2.3.3 Configuració 2

En la configuració 2 s'ha eliminat la transformació TFIDF per només tenir en compte la freqüència de paraules.

Filtre	Paràmetre	Valor
StringToWordVector	Stemmer	LovinsStemmer
	Tokenizer	WordTokenizer
	TFIDF/Freqüència de paraules	Freqüència de paraules
	StopWords	Rainbow
AttributeSelection	InfoGainAttributeEval	>0

Taula 48. Configuració de proves 2

6.2.2.3.4 Configuració 3

En aquesta tercera configuració, s'ha intentat eliminar el diccionari de StopWords per comprovar com afectaven les paraules brossa al resultat ofert per l'algorisme.

Filtre	Paràmetre	Valor
StringToWordVector	Stemmer	LovinsStemmer
	Tokenizer	WordTokenizer
	TFIDF/Freqüència de paraules	Freqüència de paraules
	StopWords	Cap
Selecció d'atributs	InfoGainAttributeEval	>0

Taula 49. Configuració de proves 3

6.2.2.3.5 Configuració 4

En la penúltima configuració, s'ha eliminat el llindar de guany d'informació. Aquesta acció ha provocat que tots els atributs inicials del dataset s'hagin mantingut en el dataset final al no realitzar cap tipus de filtratge.

Filtre	Paràmetre	Valor
StringToWordVector	Stemmer	LovinsStemmer
	Tokenizer	WordTokenizer
	TFIDF/Freqüència de paraules	TFIDF
	StopWords	Rainbow
Selecció d'atributs	InfoGainAttributeEval	0

Taula 50. Configuració de proves 4

6.2.2.3.6 Configuració 5

En l'última configuració s'ha decidit utilitzar *ngrams*, és a dir, conjunts de paraules com atribut en comptes de tenir-ne una única per cada característica del dataset. Aquest mètode ha donat bons resultats en alguns estudis basats en la classificació de textos (Cavnar et al,.1994).

Filtre	Paràmetre	Valor
StringToWordVector	Stemmer	LovinsStemmer
	Tokenizer	NGramTokenizer
	TFIDF/Freqüència de paraules	TFIDF
	StopWords	Rainbow
Selecció d'atributs	InfoGainAttributeEval	>0

Taula 51. Configuració de proves 5

6.2.2.3.7 Resultats de les configuracions

Tal com el lector pot comprovar a la taula 52, la configuració 1 aparenta ser la més adient entre totes les realitzades. Per una banda, el resultat dels corpus de prova és aleatori, el qual implica que és indiferent el seu resultat al no ser acceptable com a model d'entrenament. Tot i

així, la tècnica de validació creuada mostra un resultat del 82.2% en la primera configuració, un resultat superior en comparació a les altres proves realitzades.

Configuració	Tipus de validació	Instàncies correctament classificades
Configuració 1	Dataset de proves	48%
	Validació creuada	82.2%
Configuració 2	Dataset de proves	46.5%
	Validació creuada	80.9%
Configuració 3	Dataset de proves	48%
	Validació creuada	79.6%
Configuració 4	Dataset de proves	48.5%
	Validació creuada	78.3%
Configuració 5	Dataset de proves	49%
	Validació creuada	69.9%

Taula 52. Resultats de la configuracions de proves

Un cop seleccionada la configuració, cal aplicar aquesta a tots els datasets creats en els passos anteriors, gràcies a això s'obtindran una sèrie de arxius ARFF nous que correspondran amb les transformacions realitzades. És important recordar al lector que per cada subconjunt d'instàncies creat, existirà l'arxiu ARFF corresponent al dataset de proves amb els mateixos atributs que el primer. La taula 53 s'utilitza a mode de resum per tal de facilitar la comprensió d'aquest fet.

Arxiu	Descripció
trainDataSetXX.arff	Arxiu ARFF amb el dataset corresponent a XX instàncies.
trainDataSetXX.vector.arff	Arxiu ARFF creat un cop convertides les instàncies a un vector de característiques.
trainDataSetXX.vector.final.arff	Arxiu ARFF final creat a partir del procés de selecció d'atributs mitjançant l'arxiu anterior.
testDataSet.arff	Arxiu ARFF amb el dataset de proves original.
testDataSetXX.vector.arff	Arxiu ARFF creat un cop convertides les instàncies a un vector de característiques. Gràcies al procés de batch-filtering contindrà els mateixos atributs que l'arxiu trainDataSetXX.
testDataSetXX.vector.final.arff	Arxiu ARFF final creat a partir del procés de selecció d'atributs. Utilitzant el procés de batch-filtering, els atributs eliminats seran els mateixos que els corresponents a l'arxiu trainDataSetXX.

Taula 53. Nomenclatura dels arxius creats per la experimentació

A continuació, abans de explicar els passos a realitzar per tal d'obtenir els valors de l'experimentació, es descriuran les mètriques a utilitzar per valorar el rendiment dels algorismes a utilitzar.

6.2.3 Anàlisis de resultats obtinguts

Per tal d'analitzar els diferents resultats obtinguts de l'experimentació s'utilitzaran una sèrie de mètriques proporcionades per WEKA les quals podran ajudar a saber si els algoritmes utilitzats són viables.

Les mètriques utilitzades durant l'experimentació són les següents:

- **Percentatge d'instàncies categoritzades correctament**: El primer factor per analitzar els algoritmes és el percentatge que aquest aconsegueix classificar correctament. Tot i que és una bona mètrica per analitzar el seu rendiment en brut, no deix de ser relativament superficial en l'anàlisi d'algoritmes.
- **Precisió**: La precisió és una mètrica que permetrà saber si realment una classe ha estat ben classificada o no. La formula 12 permet calcular aquesta mètrica de forma manual.

$$Precision = \frac{tp}{tp + fp}$$

Formula 12. Formula de la precisió

- **Recall**: Aquesta mètrica ens permetrà indicar quants cops una instància ha estat classificada a la classe correcta i no confosa per un altre. La fórmula 13 mostra al lector com es realitza el càlcul d'aquesta mètrica.

$$Recall = \frac{tp}{tp + fn}$$

Formula 13. Formula de recall

- **F-Measure**: La última mètrica és una combinació de les mètriques anteriors. Representa la intersecció entre la precisió i el recall, normalitzat mitjançant la suma de les dues. La fórmula 14 mostra proporciona al lector la seva expressió.

$$F\text{-}measure = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Formula 14. Formula de la F-measure

A més de les descrites anteriorment, WEKA ofereix altres mètriques per tal de mostrar el rendiment dels algorismes, tot i així, s'han escollit les anteriors al ser mètriques ja utilitzades amb èxit en altres estudis de classificació de textos d'un àmbit similar al realitzat en el projecte (Vazquez, Muñoz-Garcia, Campanella, Poch, Fisas, Bel i Andreu et al,. 2014).

Tot i que durant l'experimentació s'utilitzaran totes les mètriques descrites, prioritzar el valor d'una mètrica sobre l'altra pot ser útil segons l'objectiu que es vulgui aconseguir. Prioritzar el recall sobre la precisió pot causar que un algorisme augmenti els falsos positius fent disminuir la precisió. Per altra banda, prioritzar la precisió pot provocar que molts positius es perdin al tenir un llindar d'acceptació de positius massa elevat. Un exemple sobre la necessitat de donar més importància a una mètrica sobre un altre es podria donar en el cas de cercar un algorisme que intenti detectar els portadors d'una malaltia contagiosa. En aquest punt podria ser preferible equivocar-se amb un fals positiu que no pas deixar a un pacient contagiós lliure per no haver sigut capaç de detectar-lo. Finalment, la mètrica F-measure permet oferir una mesura de rendiment més general al ser una mètrica combinada entre les dos anteriors, el qual pot ser una bona alternativa en cas de no requerir cap preferència.

Un cop mostrades les mètriques a utilitzar, el següent punt mostrarà la descripció del mètode a utilitzar per realitzar la obtenció de valors de l'experimentació.

6.2.4 Realització dels experiments

L'últim pas a realitzar en l'experimentació abans de realitzar el propi l'anàlisi de resultats, és el propi procés d'obtenció d'aquests.

Per poder realitzar aquest pas s'han utilitzat dos eines: la interfície d'usuari de WEKA per realitzar l'obtenció de resultats i l'aplicació Excel per anotar i organitzar els valors trobats.

Amb la utilització d'aquestes, s'ha realitzat un seguit de passos per tal d'analitzar el rendiment dels algoritmes en cadascun dels datasets creats en etapes anteriors. Un cop realitzats aquests per tots els subconjunts creats dels tres datasets d'experimentació, s'han validat els resultats mitjançant alguna de les dos tècniques disponibles. A continuació els següents subapartats descriuen breument els passos realitzats.

6.2.4.1 Selecció de l'arxiu d'entrada

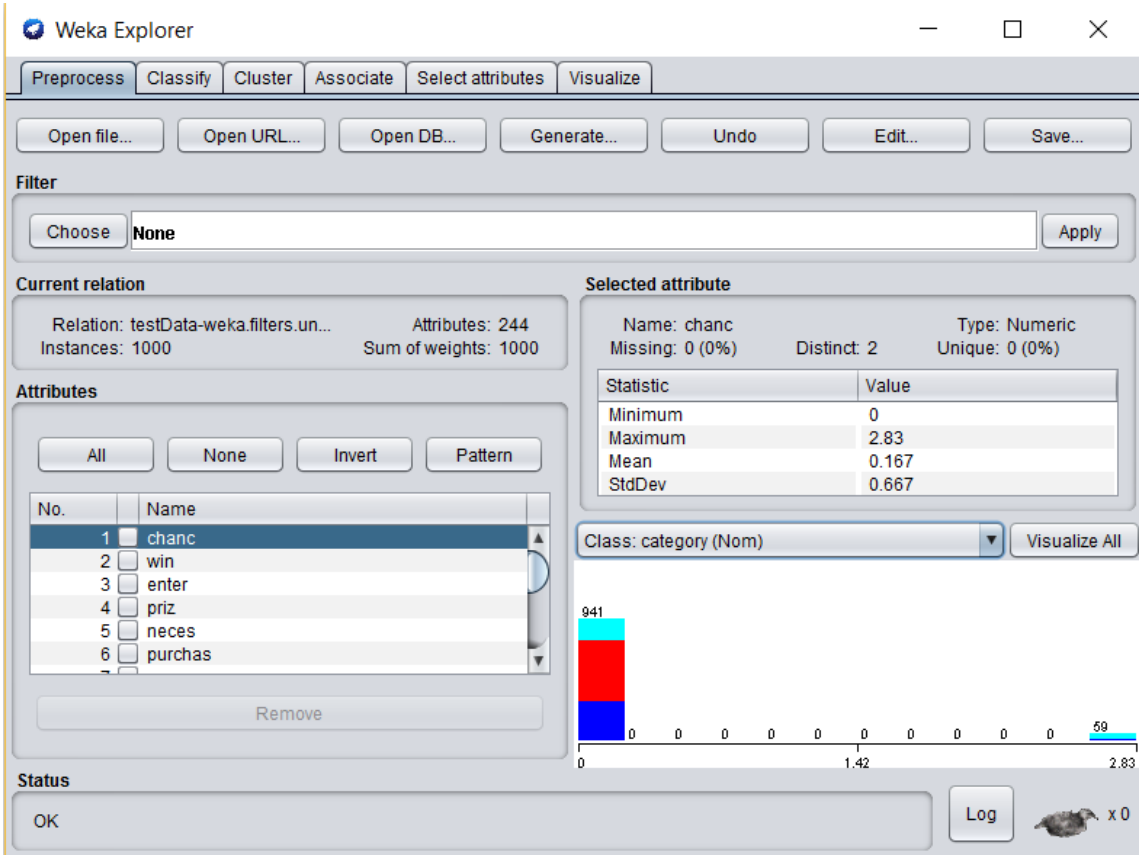
El primer pas a realitzar és seleccionar un subconjunt d'un dels corpus que es vol analitzar mitjançant la comanda 'Open file' de WEKA. Aquest arxiu com ja s'ha comentat en el punt 5.2.2.3.7 del document, segueix la nomenclatura de 'trainDataSetXX.vector.final.arff'.

6.2.4.2 Selecció de l'algorisme

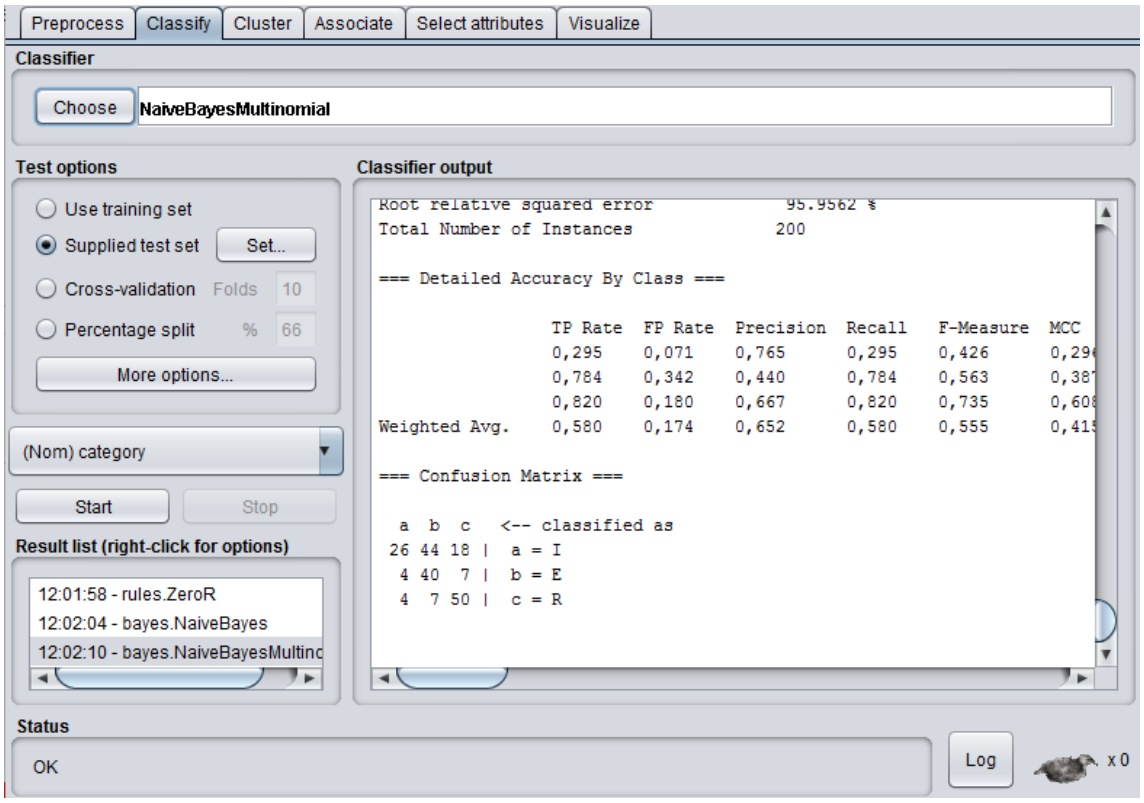
Un cop s'ha proporcionat un arxiu d'entrada, es pot procedir a seleccionar els algorismes a utilitzar per l'experimentació.

En aquest projecte, com ja s'ha esmentat en apartats anterior s'han utilitzat quatre algorismes per realitzar l'entrenament: Naive Bayes utilitzant la seva versió multinomial, al donar millors resultats que la versió bàsica de l'algorisme (McCallum i Niggam et al,. 1998), Random Forest, KNN, anomenat IBk a WEKA i finalment l'algorisme de SVM, utilitzant el nom de SMO a l'aplicació.

La selecció de l'algorisme a WEKA es realitza a pestanya 'Classify', el qual té un apartat per seleccionar el classificador desitjat. En aquest punt es pot realitzar la configuració particular de l'algorisme, però degut a que els canvis de valors realitzats no han proporcionat diferències substancials en els resultats de les proves, s'ha decidit mantenir els valors per defecte d'aquests.



Imatge 47. Pantalla de preprocessament a WEKA



Imatge 48. Pantalla de classificació d'instàncies a WEKA

6.2.4.3 Selecció d'opcions de proves

Un cop seleccionat l'algorisme, cal seleccionar el mètode de validació mitjançant la opció de 'Test options' de WEKA visible en la imatge anterior.

Durant una primera etapa del projecte, l'única opció de validació a considerar era la opció de validació mitjançant un dataset de proves. Malauradament, degut als resultats obtinguts la experimentació amb aquestes, s'ha decidit ampliar el nombre de mètodes de validació afegint la utilització de la tècnica de la validació creuada al circuit de proves.

6.2.4.4 Execució i anotació de resultats

L'últim pas a realitzar durant la experimentació, ha consistit en la execució dels algorismes amb el botó corresponent i la anotació de les dades resultants de l'execució d'aquestes a un arxiu Excel, el qual s'ha utilitzat per realitzar sèrie de gràfiques comparatives per l'anàlisi de resultats.

6.3 Resultats obtinguts

Com ja s'ha comentat en un punt anterior d'aquest projecte, els resultats obtinguts mitjançant la validació amb un corpus de proves han sigut molt negatius, arribant a obtenir resultats aleatoris amb els datasets emprats. Amb aquesta informació present i amb el motiu d'obtenir més dades dels algorismes, es va decidir ampliar la experimentació incorporant la tècnica de validació creuada en el conjunt de proves realitzats.

En els següents subapartats es mostrarà al lector l'anàlisi dels resultats de cada experimentació realitzada.

6.3.1 Diferències de rendiment entre tècniques de validació

La incorporació de la tècnica de validació creuada a l'experimentació, va aconseguir presentar millores importants en els datasets B i C. Per tal d'il·lustrar aquest fet, a continuació es mostraran al lector un conjunt de gràfiques comparatives amb el nombre d'instàncies encertades per cada dataset mitjançant les dues tècniques.

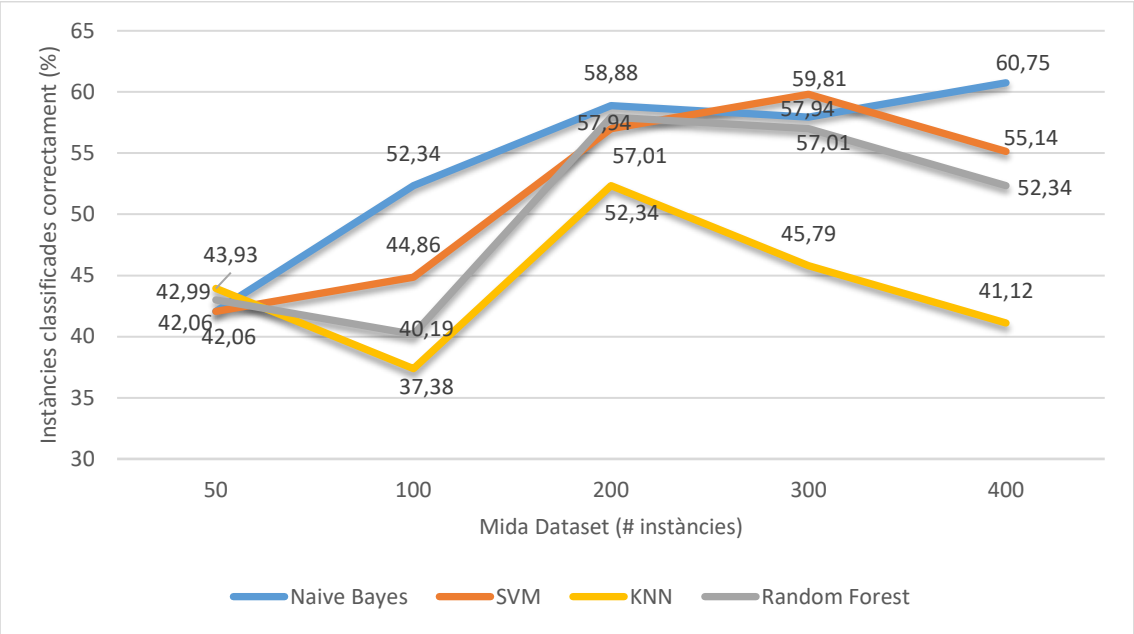
6.3.1.1 Dataset A

Les primeres dos gràfiques mostren l'evolució dels algorismes segons s'incrementa el nombre d'instàncies. A diferència dels altres dos datasets, cap de les dos tècniques de validació donen resultats vàlids.

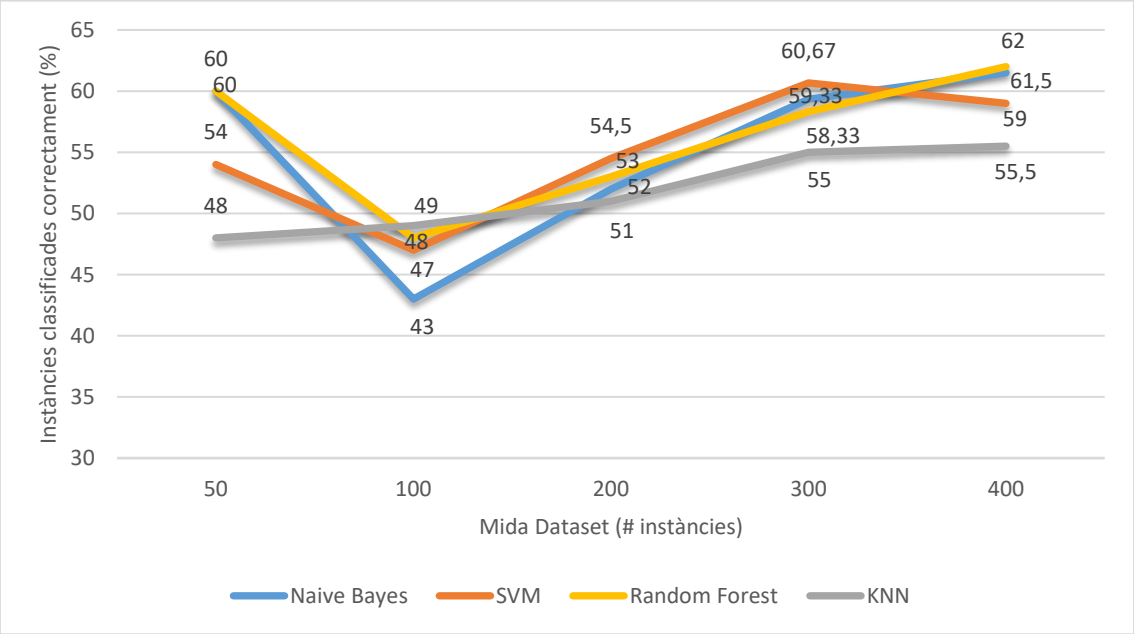
En aquest punt és interessant observar les diferències entre els datasets. Mentre que els corpus B i C tenen tres categories a classificar, el dataset ofert per la doctora Pletikosa conté quatre categories. Aquesta categoria addicional conjuntament amb el poc nombre d'instàncies disponibles, provoca que cada categoria tingui un total de menys instàncies per classes per tal de realitzar l'aprenentatge, el qual pot provocar aquests resultats.

Per altra banda també es pot observar un característica important diferencial amb els altres datasets. Tot i que els resultats no són acceptables per poder construir un model vàlid per classificar noves instàncies, els resultats generals de la validació amb el corpus de proves són lleugerament superiors als obtinguts pels altres corpus. Aquests resultats semblen indicar que les instàncies aparegudes en el conjunt de la doctora Pletikosa representen un coneixement

més general respecte a les classes que es volen classificar. Per aquest motiu, un nombre menor de instàncies ofereix més informació que un nombre major però amb més soroll incorporat en forma d'atributs.



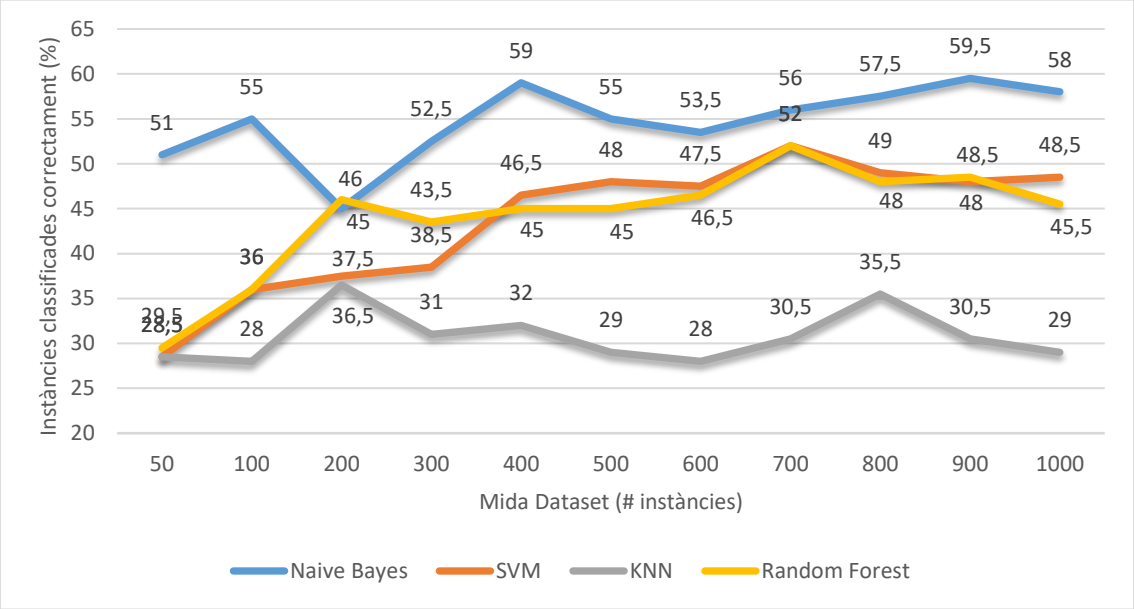
Gràfica 4. Percentatge d'encerts del dataset A utilitzant el dataset de proves



Gràfica 5. Percentatge d'encert del dataset A utilitzant validació creuada

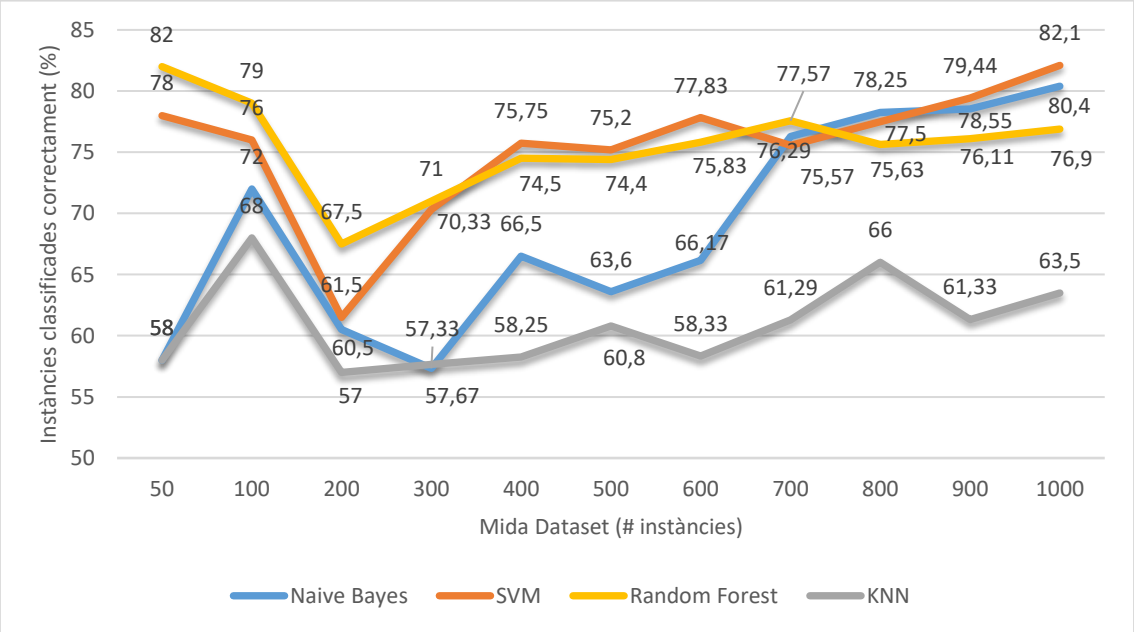
6.3.1.2 Dataset B

Les següents gràfiques mostren els resultats de les diferències entre el tipus de validació apareguts durant l'experimentació del corpus sense informació del sector al qual corresponen les dades extretes.



Gràfica 6. Percentatge d'encerts del dataset B utilitzant el dataset de proves

La gràfica 6 mostra l'evolució dels algorismes utilitzant la validació mitjançant un dataset de proves. Com ja s'ha comentat anteriorment, els resultats oferts pels algorismes són aleatoris al no arribar al 50% d'encert a excepció del Naive Bayes, fet sorprenent tenint en compte que és l'algorisme més antic i ser habitualment el que ofereix pitjors resultats en els diversos estudis consultats.



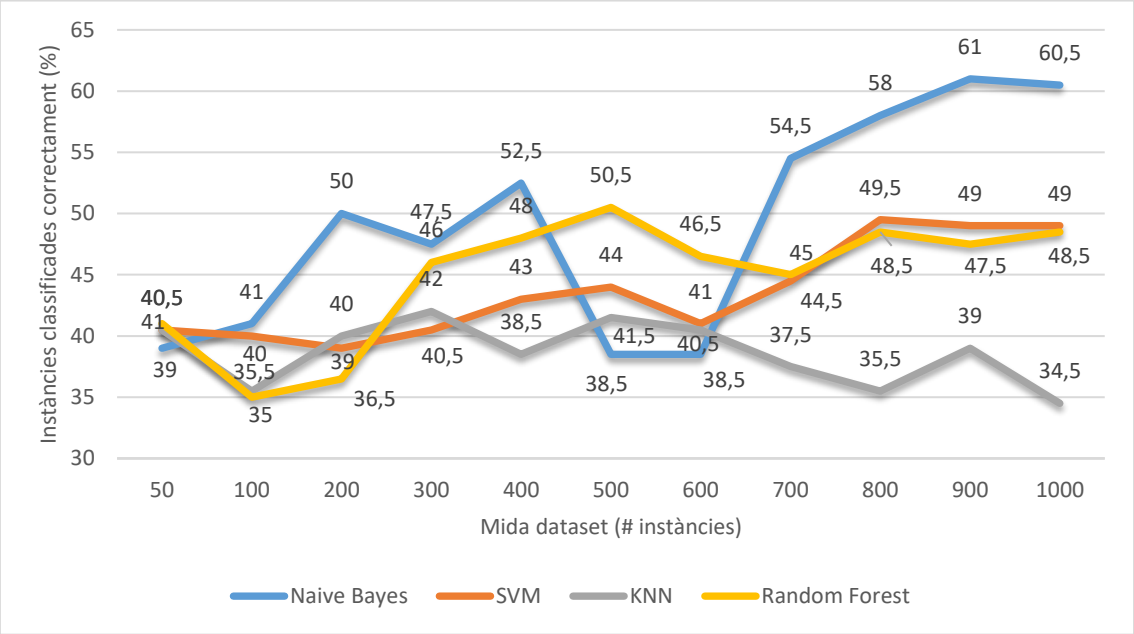
Gràfica 7. Percentatge d'encert del dataset B utilitzant validació creuada

La gràfica 7 representa la experimentació realitzada mitjançant validació creuada. Les dades ofereixen una gran millora comparada amb la validació mitjançant el dataset de proves. Es pot observar que excepte en els primers datasets amb menys nombre d'instàncies, el rendiment dels algorismes ascendeix fins un encert superior al 80% en els casos de Naive Bayes i SVM, un rendiment que ja es pot arribar a considerar acceptable. L'única excepció apareix en l'algorisme KNN, el qual no dona no proporciona un bon rendiment en cap dels casos observats.

6.3.1.3 Dataset C

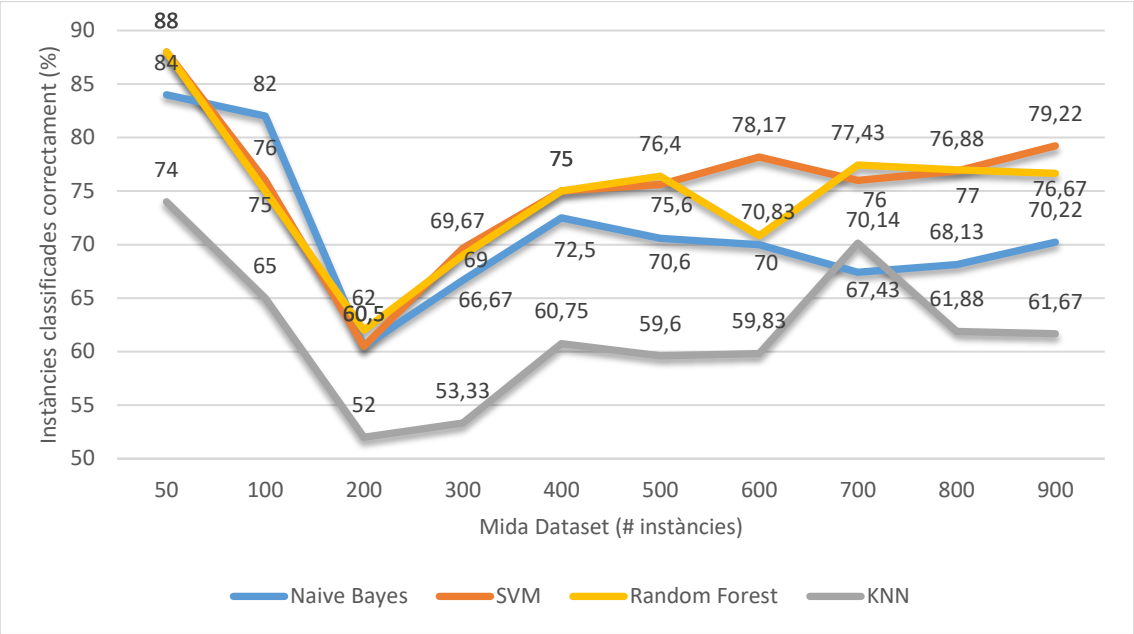
L'últim corpus a analitzar incorpora la informació del sector al conjunt d'atributs que contenen les instàncies d'entrada.

Els resultats obtinguts utilitzant la validació mitjançant un conjunt de dades de prova són similars als provinents del dataset B. Tots els algorismes, a excepció del Naive Bayes no arriben a obtenir models vàlids per una utilització posterior al no passar tan sols el llindar del 50%. L'única diferència observable és una mínima millora dels resultats en comparació amb el mateix gràfic del segon dataset. Malauradament, aquesta millora és massa marginal com per poder assegurar que la informació del sector ha ajudat a l'algorisme a millorar els seus resultats.



Gràfica 8. Percentatge d'encerts del dataset C utilitzant el dataset de proves

Aquesta segona gràfica presenta els resultats obtinguts mitjançant validació creuada. Com es pot comprovar, a diferència del cas anterior els resultats obtinguts són lleugerament inferiors als del dataset B, no arribant a superar el llindar del 80% en cap dels casos. L'algorisme SVM continua sent l'algorisme que ofereix millor rendiment en la experimentació, tot i així, sembla que l'adició de la informació del sector ha afectat el rendiment de l'algorisme Naive Bayes, el qual ha reduït el seu encert fins a 6 punts en el cas de l'entrenament amb 1000 instàncies.



Gràfica 9. Percentatge d'encert del dataset C utilitzant validació creuada

Com el lector podrà comprovar fàcilment, la diferència entre les dos tècniques de validació en aquest últim dataset és similar a la succeïda en el dataset B el qual indica que la adició del nou atribut no ha suposat cap millora a la problemàtica existent.

Per finalitzar les diferències de rendiment entre les dos tècniques de validació, el següent punt analitza el motiu de les diferències aparegudes en l'experimentació.

6.3.1.4 Sobre-especialització del model

Un cop analitzats els resultats dels dos tipus de validació, és necessari conèixer el motiu de la diferència de rendiment entre aquestes. Aquesta diferència de rendiment es deguda al *overfitting* o sobre-especialització, és a dir, el model creat mitjançant l'entrenament no ha sigut capaç de generalitzar de forma suficient els conceptes adquirits mitjançant les dades d'entrada per tal de poder categoritzar-ne de noves correctament. Per tal de minimitzar aquesta diferència existeixen diverses solucions, dos de les quals es presenten a continuació:

- **Configuració del model**: La primera solució possible primer és el canvi de paràmetres durant la creació del model. Aquest pas s'ha realitzat a l'inici de l'experimentació al veure els primers resultats tal com es pot observar en el punt 5.2.2.3 d'aquest document. Malauradament, tot i provar les diverses configuracions mostrades, cap d'elles ha sigut capaç de millorar el rendiment dels algorismes en la validació amb datasets de proves.
- **Augment dels conjunts d'entrenament**: Degut a la complexitat dels conceptes a categoritzar i tenint en compte que existeixen estudis de Sentiment Analysis en els quals els conjunts d'entrada utilitzats superaven les 10000 instàncies (Socher, Perelygin, Wu, Chuang, Manning, Y. Ng i Potts et al., 2013), des d'un inici es podia arribar a intuir que el nombre de mil instàncies d'entrenament podria no ser suficient per tal de realitzar correctament la experimentació. Un augment d'aquest nombre d'instàncies per augmentar el conjunt de conceptes que representen una classe podria ajudar a minimitzar aquest problema.

Per finalitzar, és important remarcar que el fet de requerir la utilització d'aquest tipus de validació al no obtenir resultats positius amb els corpus de prova implica des d'un inici que els resultats obtinguts en aquest tipus de validació no és extrapolable a un conjunt de dades completament independent, doncs les instàncies del propi corpus d'entrenament tenen característiques comunes entre elles. Per exemple, la persona que ha escrit el post o el període de temps en que s'ha redactat aquest són factors comuns entre moltes de les instàncies contingudes en el dataset de proves.

6.3.2 Avaluació dels algorismes amb validació creuada

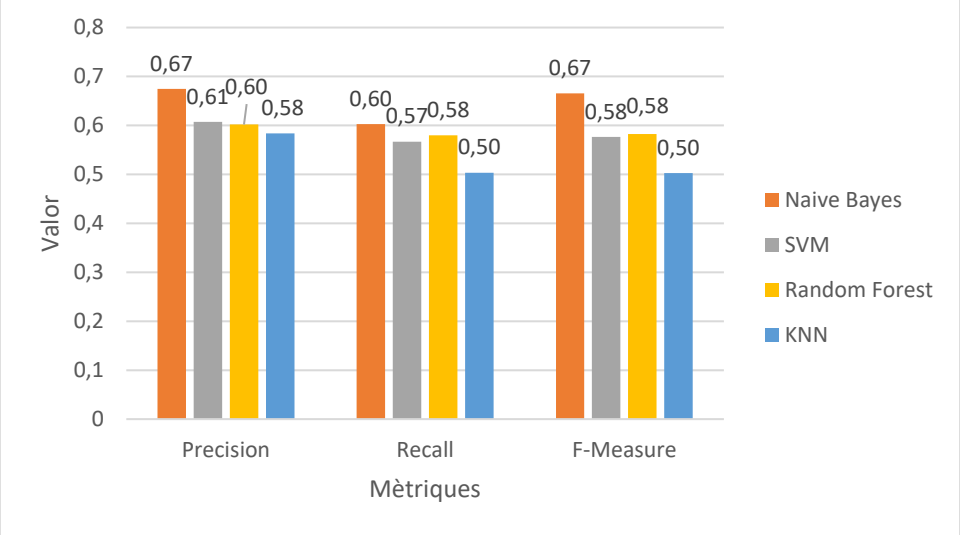
Donat que l'avaluació mitjançant el corpus de proves no ha sigut positiva, s'ha decidit continuar l'experimentació analitzant en més profunditat el rendiment dels algorismes mitjançant l'avaluació creuada utilitzant en aquest cas les mètriques de precisió, recall i F-measure mostrades en l'apartat 6.2.3 del present document.

En aquest anàlisi es mostraran els resultats obtinguts pels tres subconjunts amb el màxim d'instàncies de cadascun dels corpus emprats. És a dir, s'utilitzaran els subconjunts de 400 instàncies en el cas del dataset A i els de 1000 instàncies en el cas dels datasets B i C.

6.3.2.1 Dataset A

La gràfica 10 mostra el rendiment dels algorismes en el dataset ofert per la doctora Pletikosa.

En aquest es pot observar com l'algorisme Naive Bayes supera els altres tres algorismes amb cert marge de diferència mitjançant qualsevol de les tres mètriques. L'algorisme SVM, tot i tenir una major precisió que Random Forest, aquest últim el supera en les mètriques de Recall i F-measure, el qual indica que Random Forest té més possibilitats de detectar correctament una classe. Tot i així, degut a la seva menor precisió és possible que el positiu detectat sigui realment fals. En última instància es té l'algorisme KNN. La seva precisió és propera a la dels algorismes SVM i Random Forest, però al obtindré menys puntuació en la mètrica de recall, té menys possibilitats de detectar els positius d'una classe.

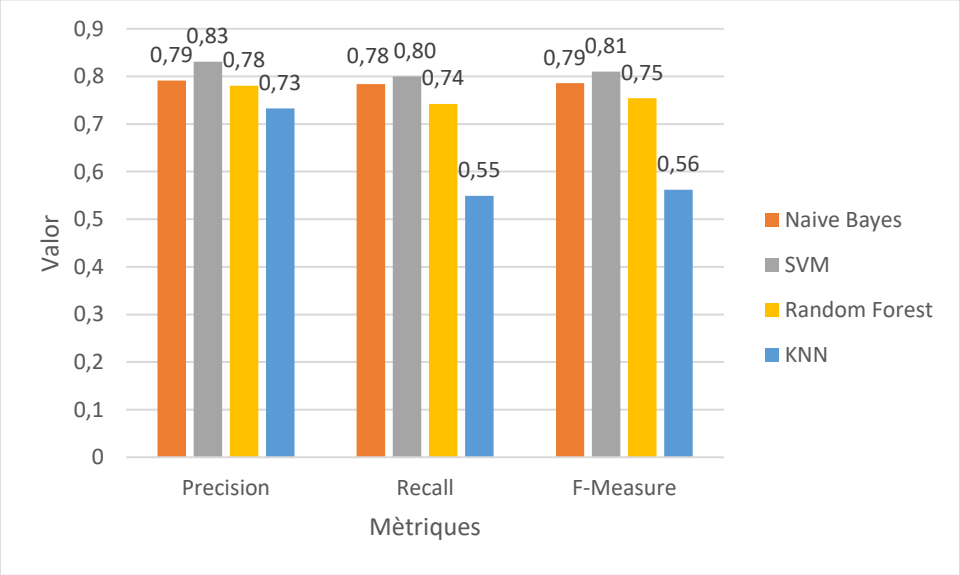


Gràfica 10. Mètriques d'avaluació del dataset A

6.3.2.2 Dataset B

El segon dataset avaluat és el dataset creat per la realització del projecte sense informació dels sectors.

En aquest corpus, l'algorisme amb millor avaluació és el SVM, superant les tres mètriques als altres tres candidats amb valors superiors al 80%. L'algorisme Naive Bayes mostra resultats lleugerament inferiors, tot i no poder superar el llindar del 80%, ofereix millor rendiment, sobretot en precisió que l'algorisme Random Forest. Aquest últim té més problemes per tal de detectar correctament una classe que els dos primers candidats. En última posició es situa de nou l'algorisme KNN, tot i que en cas de detectar una classe obté un 73% de possibilitat de ser un autèntic positiu, el seu rendiment a l'hora de detectar-les és realment pobre.

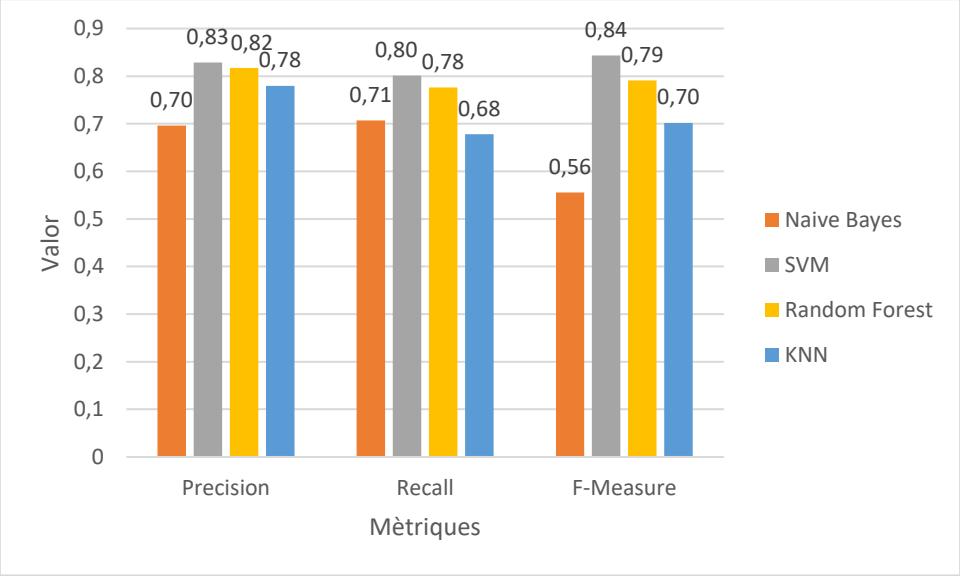


Gràfica 11. Mètriques d'avaluació del dataset B

6.3.2.3 Dataset C

L'últim dataset a analitzar és el dataset amb informació dels sectors.

En aquest últim corpus, es pot comprovar com la informació del sector ha empitjorat de forma important el rendiment de l'algorisme Naive Bayes, situant-lo en última posició respecte els altres tres algorismes. En el cas actual, el SVM continua sent l'algorisme en millor rendiment, superant amb les tres mètriques a Random Forest. Tot i així, la nova informació incorporada al corpus també ha afectat de forma lleugerament negativa al rendiment d'aquests dos algorismes. L'únic cas de millora clara s'ha obtingut amb l'algorisme KNN, el qual els valors de les mètriques han augmentat globalment, sobretot la de recall i en conseqüència, la mètrica combinada F-Measure.



Gràfica 12. Mètriques d'avaluació del dataset C

6.4 Conclusions de l'experimentació

Per oferir les conclusions d'aquesta experimentació és necessari recordar dos dels objectius que es perseguien durant aquest projecte, és a dir, comprovar la possibilitat d'utilitzar algorismes de Machine Learning per tal de categoritzar les comunicacions de les companyies a les xarxes socials i intentar conèixer el nombre d'instàncies a partir dels quals els algorismes no milloraven el seu rendiment.

Sobre la viabilitat d'utilitzar algorismes de Machine Learning i el nombre d'instàncies a partir del quals els algorismes entraven en rendiment decreixent, tot i que la validació creuada ha ofert bons resultats, la validació mitjançant els datasets de proves no s'ha pogut concloure de forma satisfactòria requerint d'un estudi amb més profunditat per tal d'obtenir resultats positius. Malauradament, per obtindré la resposta a la pregunta de la viabilitat, la possibilitat d'obtenir resultats viables mitjançant un corpus de proves és condició *sine qua non*. En el cas del nombre d'instàncies mínimes necessàries la conclusió és similar a l'anterior, doncs és necessari obtenir un resultat positiu abans de poder extreure una conclusió raonable.

Per altra banda, a l'hora d'avaluar els algorismes mitjançant la tècnica de la validació creuada s'ha observat que, sorprenentment, l'algorisme Naive Bayes és l'algorisme en millors resultats en el primer dataset, mentre que els datasets B i C, l'algorisme SVM ha denotat millor capacitat d'aprenentatge que la resta. Tot i que aquest últim resultat correspon en el resultat de diversos estudis, és necessari ampliar l'anàlisi un cop s'hagi solucionat la problemàtica de la sobre-especialització per tal de poder assegurar els resultats.

Finalment, en referència a la utilització del sector en l'entrenament dels algorismes, els resultats mostrats pel dataset C permeten extreure dos conclusions. La primera és la ineficàcia d'aquest element per ajudar a resoldre el problema de sobre-especialització. Tal com es pot veure a les gràfiques de l'apartat 6.3.1.3, l'adició d'aquest atribut no provoca una millora en els resultats dels algorismes. Per altra banda, en la utilització de la validació creuada, es pot observar com aquest atribut té un efecte visible en els algorismes de Naive Bayes i KNN. Tot i

que en l'estat actual de la investigació aquest fet no té gran importància, la pot arribar a tenir en el cas de que s'arribi a obtenir un resultat positiu amb la validació amb corpus de proves. Per exemple, en el cas d'arribar-se a demostrar que els algorismes Naive Bayes i KNN són algorismes vàlids per realitzar les classificacions de les comunicacions, l'adició del sector seria una decisió crítica a prendre abans de realitzar qualsevol tipus d'entrenament.

7 Conclusions del projecte

En aquest últim apartat del projecte, es mostraran les conclusions extretes del desenvolupament del projecte, els coneixements adquirits durant la carrera que s'han utilitzat i finalment, els possibles desenvolupaments futurs que es poden dur a terme a partir d'aquest.

7.1 Opinió de l'autor

La realització d'aquest projecte ha suposat un gran esforç personal tot i que els resultats obtinguts no han sigut tot el satisfactoris que un podria desitjar degut a un seguit de problemes que han anat apareixent durant la seva realització.

El primer problema i principal és la pròpia dedicació continua al projecte en el temps. El fet d'haver deixat de banda el projecte durant llargs períodes de temps per diversos motius, ha provocat que la realització d'aquest no hagi pogut oferir uns resultats més confiables tot i haver tingut el continu suport dels tutors del projecte per finalitzar aquest.

El segon problema trobat durant la seva realització és la evident falta de coneixements sobre la temàtica de Machine Learning i la seva utilització en la classificació de textos. És evident que el seu estudi és una part addicional de la realització del projecte, però haver obtingut certs coneixements amb anterioritat, hauria provocat un avançament molt més ràpid en la matèria.

Tot i això, en cap moment es pot considerar que hagi sigut una mala idea realitzar el projecte de final de carrera sobre aquesta temàtica. Clarament és un àmbit molt interessant i innovador en el qual tots els coneixements adquirits són de enorme valor. Els moments de descobrir nous avançaments en la tècnica de reconeixement d'imatges, veure el funcionament per primera vegada dels algorismes de classificació de textos, les facilitats que ofereixen les xarxes socials per tal de realitzar estudis de màrqueting digital, entre mols altres, han sigut certament apassionants i han provocat que els moments emprats per realitzar el projecte hagin valgut la pena.

7.2 Coneixements utilitzats

Els coneixements utilitzats durant la realització d'aquest projecte es poden dividir en dos àmbits clarament diferenciats:

- **Coneixements de programació**: Per tal de realitzar la implementació del sistema s'han utilitzat tots aquells coneixements de programació adquirits durant la carrera, tan per l'anàlisi, com per el disseny i la implementació. Assignatures com PROP, la qual va consistir en el desenvolupament d'un sistema d'inici a fi en totes les seves etapes, han ajudat a obtenir una visió global sobre el transcurs del desenvolupament. Addicionalment, han ofert coneixements que han ajudat a l'hora de realitzar l'anàlisi sobre el qual realitzar la base de l'aplicació.
- **Gestió de projectes**: Per poder realitzar les planificacions dels apartats corresponents s'han utilitzats coneixements de les assignatures de gestió de projectes realitzades a la facultat, com són PDGPE i PGPSI.
- **Recerca d'informació**: El present projecte entra dintre de l'àmbit de treball d'investigació, allunyant-se dels projectes els quals el seu objectiu consisteix en el desenvolupament d'una aplicació. Aquest fet ha provocat que la capacitat de recerca d'informació hagi sigut primordial en el transcurs d'aquest projecte.

7.3 Treball futur

Un cop finalitzat el projecte, es poden observar diverses vies sobre les quals continuar el treball realitzat.

Per l'àmbit de la recerca, és necessari continuar investigant la matèria per tal de poder obtenir millores en els resultats de l'experimentació. Aquestes millores poden esser donades mitjançant l'ampliació dels corpus d'entrenament i també per la millora de configuració dels models.

Addicionalment, degut a que gran quantitat de comunicacions a les xarxes es realitzen mitjançant imatges, es podria ampliar l'àmbit de la recerca a aquest sector per intentar incorporar-lo. Durant el transcurs del projecte es va contactar amb Andrej Karpathy, investigador de Stanford en la recerca de descripció d'imatges mitjançant Machine Learning per tal de comprovar la viabilitat d'utilitzar el seu algorisme en el projecte. La possibilitat en aquell moment va quedar descartada ja que l'algorisme encara no permetia entrenar imatges externes sense realitzar un procés previ complex, tot i així, davant de la possibilitat de que aquest fet canviï en un futur pròxim pot arribar a ser interessant contemplar aquesta possibilitat.

En l'àmbit de programació d'aplicacions, l'extractor de dades ofereix una sèrie de funcionalitats bàsiques amb molt de marge de millora, no només en l'aspecte visual, sinó també en el funcional que pot suposar un objectiu al qual enfocar-se en un futur. Addicionalment, en cas d'aconseguir obtenir bons resultats amb els datasets de proves, es pot arribar a ampliar la aplicació amb funcionalitats pròpies de WEKA, és a dir, es podria arribar a afegir la funcionalitat d'automatitzar l'entrenament, validació i fins i tot, la classificació de noves instàncies arribant a creant una suite de Machine Learning amb la qual poder realitzar estudis.

8 Bibliografia

Breiman, Leo. 2001. «Random Forests». Machine Learning 45 (1): 5-32. doi:10.1023/A:1010933404324.

Cavnar, William B., John M. Trenkle, y others. 1994. «N-gram-based text categorization». Ann Arbor MI 48113 (2): 161–175. doi:10.1.1.21.3248.

Cvijikj, Irena Pletikosa, y Florian Michahelles. 2011. «Monitoring Trends on Facebook». En , 895-902. IEEE. doi:10.1109/DASC.2011.150.

Cvijikj, Irena Pletikosa, y Florian Michahelles. 2013. «Online Engagement Factors on Facebook Brand Pages». Social Network Analysis and Mining 3 (4): 843-61. doi:10.1007/s13278-013-0098-8.

de Vries, Lisette, Sonja Gensler, y Peter S.H. Leeflang. 2012. «Popularity of Brand Posts on Brand Fan Pages: An Investigation of the Effects of Social Media Marketing». Journal of Interactive Marketing 26 (2): 83-91. doi:10.1016/j.intmar.2012.01.003.

Espanya. Llei Orgànica 27/2014, del 27 de novembre, de Impost sobre societats. BOE, 28 de novembre de 2014, núm. 288, p. 17-18

F. Provost. 2000. «Machine learning from imbalanced data sets 10». Working Notes of the AAAI'00 Workshop on Learning from Imbalanced Data Sets, Austin, TX, pp.1–3.

Go, Alec, Richa Bhayani, y Lei Huang. 2009. «Twitter sentiment classification using distant supervision». CS224N Project Report, Stanford 1: 12.

Ho, Tin Kam (1995). Random Decision Forests (PDF). Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, 14–16 August 1995. pp. 278–282.

Joachims, Thorsten. 1997. «Text Categorization with Support Vector Machines: Learning with Many Relevant Features».

Karpathy, Andrej, y Li Fei-Fei. 2015. «Deep visual-semantic alignments for generating image descriptions». En Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,3128–3137.

Liaw, Andy, y Matthew Wiener. 2002. «Classification and regression by randomForest». R news 2 (3): 18–22.

Manzanilla, Orestes. 2015«Optimization & Machine Learning: ¿Qué son las SVM?» 2016. Accedido enero 12. http://optimachine.blogspot.com.es/2008/03/qu-son-las-svm.html.

McCallum, Andrew, Kamal Nigam, y others. 1998. «A comparison of event models for naive bayes text classification». En AAAI-98 workshop on learning for text categorization, 752:41–48. Citeseer. doi:10.1.1.46.1529.

Møller, Chr, y Milton S. Plesset. 1934. «Note on an approximation treatment for many-electron systems». Physical Review 46 (7): 618.

Moujahid, Abdelmalik, Inaki Inza, y Pedro Larranaga. 2016. «Tema 5. Clasificadores K-NN». Accedido septiembre 21. http://www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/t9knn.pdf.

Nasukawa, Tetsuya, y Jeonghee Yi. 2003. «Sentiment analysis: Capturing favorability using natural language processing». En Proceedings of the 2nd international conference on Knowledge capture, 70–77. ACM. doi:10.1145/945645.945658.

Nikitenko, Tania. «Top 5 American Pizza Brands in Social Media». 2015. Accedido mayo 14. https://www.truesocialmetrics.com/blog/top-5-american-pizza-brands

Sabate, F., Berbegal-Mirabent, J. Cañabate, A., Lebherzd, P. R. 2014. «Factors influencing popularity of branded content in Facebook fan pages». European Management Journal, 32: 1001–1011. doi:10.1016/j.emj.2014.05.001

Sayad, Saed. «Decision Tree». 2016. Accedido marzo 15. http://www.saedsayad.com/decision_tree.htm.

Simon, Phil. 2013. Too big to ignore: the business case for big data. Wiley & SAS business series. Hoboken, New Jersey: John Wiley & Sons, Inc.

Socher, Richard, Alex Perelygin, Jean Y. Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, y Christopher Potts. 2013. «Recursive deep models for semantic compositionality over a sentiment treebank». En Proceedings of the conference on empirical methods in natural language processing (EMNLP), 1631:1642. Citeseer. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.383.1327&rep=rep1&type=pdf.

Rexer, Karl. 2016. «Rexer Data Science Survey Highlights Apr-2016 - Rexer_Data_Science_Survey_Highlights_Apr-2016.pdf». 2016. Accedido agosto 17. *http://www.rexeranalytics.com/files/Rexer_Data_Science_Survey_Highlights_Apr-2016.pdf*.

Rocapal. 2016. «Playing with machine learning: Linear Regression | ¿Rocapal or Lapacor?» Accedido noviembre 11. http://blog.rocapal.org/?p=312.

Technobium. «Decision trees explained using Weka». 2016. technobium. enero 31. http://technobium.com/decision-trees-explained-using-weka/.

Turney, Peter D. 2002. «Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews». En Proceedings of the 40th annual meeting on association for computational linguistics, 417–424. Association for Computational Linguistics. doi:10.3115/1073083.1073153.

Tumasjan, Andranik, Timm Oliver Sprenger, Philipp G. Sandner, y Isabell M. Welpe. 2010. «Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment.» ICWSM 10: 178–185.

Vázquez, Silvia, Óscar Muñoz-García, Inés Campanella, Marc Poch, Beatriz Fisas, Nuria Bel, y Gloria Andreu. 2014. «A Classification of User-Generated Content into Consumer Decision Journey Stages». Neural Networks 58 (octubre): 68-81. doi:10.1016/j.neunet.2014.05.026.

Viquipèdia. «Validació creuada». 2016. Viquipèdia, l'enciclopèdia lliure. https://ca.wikipedia.org/w/index.php?title=Validaci%C3%B3_creuada&oldid=16847938.

Witten, I. H., Eibe Frank, y Mark A. Hall. 2011. Data mining: practical machine learning tools and techniques. 3rd ed. Morgan Kaufmann series in data management systems. Burlington, MA: Morgan Kaufmann.

Yang, Yiming, y Pedersen, Jan O. 1997. «A comparative study on feature selection in text categorization». doi:10.1.1.32.9956.

Yang, Yiming, y Xin Liu. 1999. «A re-examination of text categorization methods». En Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, 42–49. ACM. http://dl.acm.org/citation.cfm?id=312647.

Zhang, Zhu, y Xi Lin. s. f. «Controversy is Marketing: Mining Sentiments in Social Media».